ParaGEMS: Integrating discrete exterior calculus (DEC) into ParaFEM for geometric analysis of solid mechanics

Document Version
Final published version

[Link to publication record in Manchester Research Explorer](#)

OPEN ACCESS

Original software publication

ParaGEMS: Integrating discrete exterior calculus (DEC) into ParaFEM for geometric analysis of solid mechanics

Pieter D. Boom *,1, Andrey P. Jivkov, Lee Margetts

School of Engineering, University of Manchester, George Begg Building, Manchester, M1 3BB, UK

ARTICLE INFO

ABSTRACT

New high-performance computing (HPC) software designed for massively parallel computers with high-speed interconnects is presented to accelerate research into geometric formulations of solid mechanics based on discrete exterior calculus (DEC). DEC is a relatively new and entirely discrete approach being developed to model non-smooth material processes, for which continuum descriptions fail. Until now, progress has been slowed by limited HPC software. The tool presented herein integrates the DEC library ParaGEMS into the well-established parallel finite-element (FE) code ParaFEM, leveraging ParaFEM's diverse IO routines, optimised solvers, and interfaces to third-party libraries. This is accomplished by interpreting FE elements, or their subdivision, as independent DEC simplicial complexes. The element-wise contribution to the global system matrix is then replaced with the DEC formalism, superimposing contributions from the dual mesh at element boundaries. The integrated tool is validated using five miniApps for scalar diffusion and linear elasticity on synthetic microstructures with emerging discontinuities, showing the performance for both continuum and discrete problems. Profiling indicates DEC calculations have excellent scaling and the solver achieves approximately 80% parallel efficiency using naïve partitioning on ~8000 cores with >135 million unknowns. The tool is now being used to develop DEC formulations of more complex phenomena, such as material nonlinearity and fracture.

Code metadata

Current code version	*ParaGEMS (2022.02.28); ParaFEM (5.0.3)*
Permanent link to code/repository used for this code version	https://github.com/ElsevierSoftwareX/SOFTX-D-22-00181
Permanent link to reproducible capsule	
Legal code license	*BSD 2-clause*
Code versioning system used	*git*
Software code languages, tools and services used	*Fortran, MPI, BLAS, LAPACK*
Compilation requirements, operating environments and dependencies	*Fortran90; MacOS, Linux, Windows Subsystem for Linux (WSL)*
If available, link to developer documentation/manual	https://github.com/ParaFEM/ParaGEMS/README.md; https://github.com/ParaFEM/ParaGEMS/lib_paragems/docs
Support email for questions	pieter.boom@kfupm.edu.sa, lee.margetts@manchester.ac.uk

1. Motivation and significance

Traditional engineering scale simulation in solid mechanics is built on a continuum assumption: that properties and processes within a material vary continuously in space. This works well for modelling smooth macroscopic behaviours; however, by definition, continuum approaches cannot fundamentally represent discrete or discontinuous features. Many materials have a discrete structure, even above the atomic level. This is obvious in some cases, like for fibre-reinforced plastics or concrete, but still exist in other materials like polycrystalline metals. For example, observe the discrete cellular grain structure of an iron-carbon alloy at the mesoscale; Fig. 1 (left) [1]. These structures influence
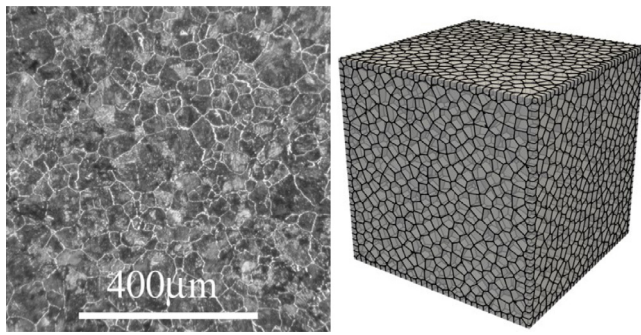

Fig. 1. Grain structure of an iron-carbon alloy (left); synthetic grain structure (right). Source (left): www.doitpoms.ac.uk/tlplib/atomic-scale-structure/poly.php?printable=1 [1].

emergent behaviours, like thermal and chemical transport, as well as non-smooth processes like fracture.

Accurately modelling the discrete structure of materials, and the non-smooth processes within, is critically important to predict how they will ultimately degrade and fail. As materials are the building blocks of both common and critical high-valued infrastructure, failure to accurately predict non-smooth behaviours in solid mechanics 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.

In contrast to continuum approaches, discrete exterior calculus (DEC) is built on a fundamentally discrete view of the world and can mimic the discrete structures in nature; Fig. 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 [3]. The connections form maps, interpreted as exterior derivatives, that enables common vector calculus operations to be mimicked [3]. This makes DEC appealing for simulating discrete structure and non-smooth physical processes across length scales from the molecular (micro) through to the engineering (macro).

Interest in DEC is growing internationally to describe various phenomena in science and engineering [4–6]. The potential advantages for solid mechanics were observed early-on [7]; however, the first complete formulation of linear elasticity was only presented this year [2]. With a strong emphasis on mesoscale modelling, sufficiently large meshes are required to verify and validate new theory. Recently the authors developed the first parallelised DEC math library, ParaGEMS, tested for problems involving scalar transport in cracked media [8]. In the present article, ParaGEMS is integrated into the more well-established finite-element library ParaFEM [9]. The aim is to accelerate theoretical development and practical application of DEC by reusing ParaFEM's optimised parallel code for: I/O in various formats, 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 [10] or microFE for grainscale modelling. With a sustainable, robust and efficient HPC research platform designed for massively parallel computers with high-speed interconnects, 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 [8].

The FE library ParaFEM is also open-source with BSD license developed for parallel solution of various types of problems, including stress analysis of linear and nonlinear (thermo, elastic, plastic) materials, heat flow, fluid flow, eigenvalue and forced vibrations. These analyses are implemented in dozens miniapps distributed with the code and described in the textbook "Programming the Finite Element Method" [9]. Like ParaGEMS, it is written in modern Fortran with MPI parallelism and includes interfaces to BLAS, LAPACK, METIS and PETSc [11]. 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 [9,12]. Built-in solvers have been shown to have excellent performance compared with popular packages such as PETSc, and with lower memory usage [13]. ParaFEM 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, such as fracture [10], material interface joining [12], fluid–structure interaction [14], and more [15–18].

2.1. Library integration

Both FE and DEC operate on meshes of connected and conformal elements; therefore, input and output data are in similar formats and similar partitioning and load balancing approaches work well. Ultimately both methods require efficient solution of large systems of (potentially nonlinear) equations. Therefore, the approach taken to integrate DEC into ParaFEM was to modify only the mathematical contribution from each element to 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 future opportunities for coupled FE-DEC simulation within the same code.

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. 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, is 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
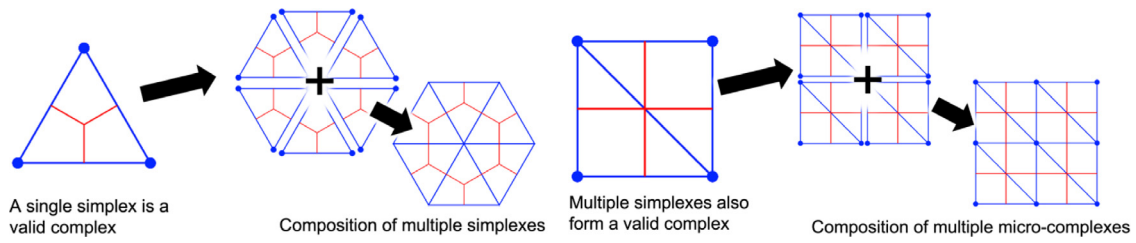
Fig. 2. The superposition of independent DEC simplexes/micro-complexes converted from FE elements used in the integrated ParaGEMS-ParaFEM miniApps.

face of the hexahedron relative to its 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

Here the dual Voronoi complex is assumed to represent the material microstructure of interest, as it can recreate more complex geometries, and the simplicial complex describes the interaction of adjacent Voronoi cells. Each cell in one complex overlaps multiple cells in the other, making a clean parallel partitioning unclear. MiniApps developed for ParaFEM partition FE elements and enforce coupling along the element interfaces [9]. In DEC terms, the ParaFEM partitioning strategy is equivalent to partitioning primal simplices (or micro-complexes for quads and hexs; Fig. 2) with coupling through the middle of Voronoi cells. Fortunately, the geometric contributions from different simplices to a single dual Voronoi cell can be superimposed. 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; Fig. 2.

3. Illustrative examples

Four miniApps were adapted from the textbook "Programming the Finite Element Method" [9] to demonstrate the integration of the libraries, maintaining the same IO, initialisation, and solver, but with different formation of the system matrix using DEC. 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, pg123x, by further modifying pg123 to iteratively zero the diffusivity of dual Voronoi faces where the gradient is above some threshold. This enables us to evaluate the performance of the method on an evolving topology, key to modelling crack initiation and growth in the future. The miniapps created correspond to problems for which there are existing DEC formulations. It is our hope that this tool will facilitate the development and testing of DEC-based description of more complex phenomena in the future.

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 to compute a diffusion coefficient (pg124/pg125); and while the simplicity of introducing discontinuities is a feature of DEC, the solution is equivalent to steady state 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. Further details are discussed briefly in a subsequent section.

To evaluate the performance of the integrated libraries, 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. The solutions were computed on the ARCHER2 supercomputer (4 cabinet system) with between 128 to 32768 cores (1 to 256 nodes). The implicit miniapps make use of an element-by-element preconditioned conjugate gradient method; however, a naïve partitioning strategy is used which assigns contiguous groups of vertices and elements, sorted by their xyz-coordinates, to successive processors. The solution vector is initialised to zero at the start of each simulation and the convergence tolerance set to 10^{-5}.

Formation of the global system matrix using DEC show excellent scaling up to 16k cores on the finest meshes with $\geq 512^3$ elements (Fig. 3 and Table 1). 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 easily fits in memory on a single compute node, meaning that large-scale simulations that densely populate the system are both possible and recover good performance. Furthermore, the trends suggest that this good performance may extend for higher core counts if larger meshes are considered.

Similarly, the solution of the final linear system achieves approximately 80% parallel efficiency with 512^3 elements on up to 8k cores before beginning to drop off (Fig. 4 and Table 2).

Table 1
Strong (top) and weak (bottom) parallel efficiency of pg123 for DEC system formation.

Mesh	CPUs							
	128	256	512	1024	2048	4096	8192	16384
128^3	100%	93%	81%	59%	35%	–	–	–
256^3	100%	99%	95%	87%	68%	51%	35%	–
512^3	100%	102%	98%	95%	86%	80%	64%	52%
600^3	100%	86%	76%	89%	70%	68%	54%	46%

| Mesh	CPUs				
	128–1024	256–2048	512–4096	1024–8192	2048–16384
128^2–256^3	87%	74%	63%	60%	–
256^2–512^3	94%	87%	83%	64%	56%

Table 2
Strong (top) and weak (middle — raw CPU timings; bottom — CPU timings normalised by solver iterations) parallel efficiency of pg123 solver.

Mesh	CPUs							
	128	256	512	1024	2048	4096	8192	16384
256^3	100%	100%	100%	89%	112%	169%	–	–
512^3	100%	90%	110%	112%	110%	97%	83%	48%
600^3	100%	110%	102%	106%	113%	133%	109%	67%

Mesh	CPUs				
	128–1024	256–2048	512–4096	1024–8192	2048–16384
128^2–256^3	35%	32%	23%	15%	–
256^2–512^3	49%	49%	43%	15%	6%

Mesh	CPUs				
	128–1024	256–2048	512–4096	1024–8192	2048–16384
128^2–256^3	67%	62%	44%	29%	–
256^2–512^3	91%	89%	79%	51%	22%
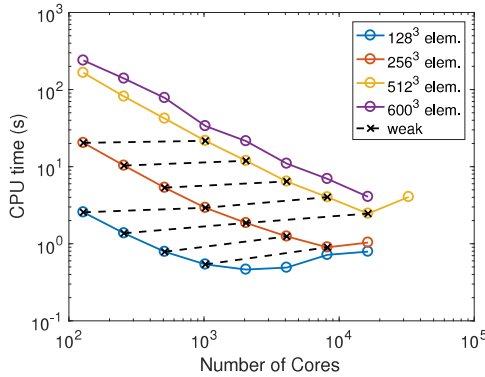
Fig. 3. Strong and weak scaling of pg123 for DEC system formation.

This is very important as it represents the bulk of the simulation time. The performance of the solver is also compared between DEC and FE formulations of the same problem. When scaled by iterations, very similar results are obtained, indicating that the DEC formulation is not having and adverse impact on the solver. Also shown is the weak scaling of the solver, which is poor when comparing raw timings, but reasonably good per iteration. This behaviour is also observed for FE simulations (not shown) and may be influenced by the naïve partitioning strategy used.

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 classical vector fields, called discrete musical isomorphisms, rather than operating solely on chain and cochain complexes. This has been approximated in the past using local Moore–Penrose pseudo inverses [2]. However, rather than mapping to and from vector fields evaluated at vertex locations, the current implementation maps to and from the midpoint of primal edges. Consider a simplex defined by vertices $s_1 = (v_1, v_2, v_3, v_4)$. Now define the augmented coboundary operator for the edge $e_1 = (v_1, v_2)$

$$\delta_{0,e\_1,aug} = \begin{bmatrix} BC_{cc}(v2) - BC_{cc}(v1) \\ BC_{cc}(v1, v2, v3) - BC_{cc}(v1, v2) \\ BC_{cc}(v1, v2, v4) - BC_{cc}(v1, v2) \end{bmatrix},$$

where BC_{cc} denotes the barycentric coordinates of the arguments' circumcentre with respect to s_1. The first row is the standard coboundary for edge e_1. The second and third rows give the coboundary of the edges between the circumcentre of e_1 to the circumcentres of the faces $f_1 = (v1, v2, v3)$ and $f_2 = (v1, v2, v4)$, respectively. These edges are related to the local part of the Voronoi face dual to edge e_1. Next define the projection operator

$$P = \delta_{0,e_1,aug} \begin{bmatrix} x\left(\begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix}\right) & y\left(\begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix}\right) & z\left(\begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix}\right) \end{bmatrix},$$

whose first row and inverse are discrete flat and sharp musical isomorphisms, respectively: $\flat = P_{1,:}$ and $\sharp = P^{-1}$. Finally, using the Hooke's law to define the relationship between the displacement gradient and stress tensor, C, we can write the local divergence of stress along the edge e_1 as

$$(\delta_{e_1}^T \star_1 \flat \otimes I_3)C(\sharp \delta_{0,e_1,aug} \otimes I_3).$$

This procedure allows calculations of the contribution from the current primal element to stay entirely local, and then be summed with adjacent elements as described earlier. Another benefit is
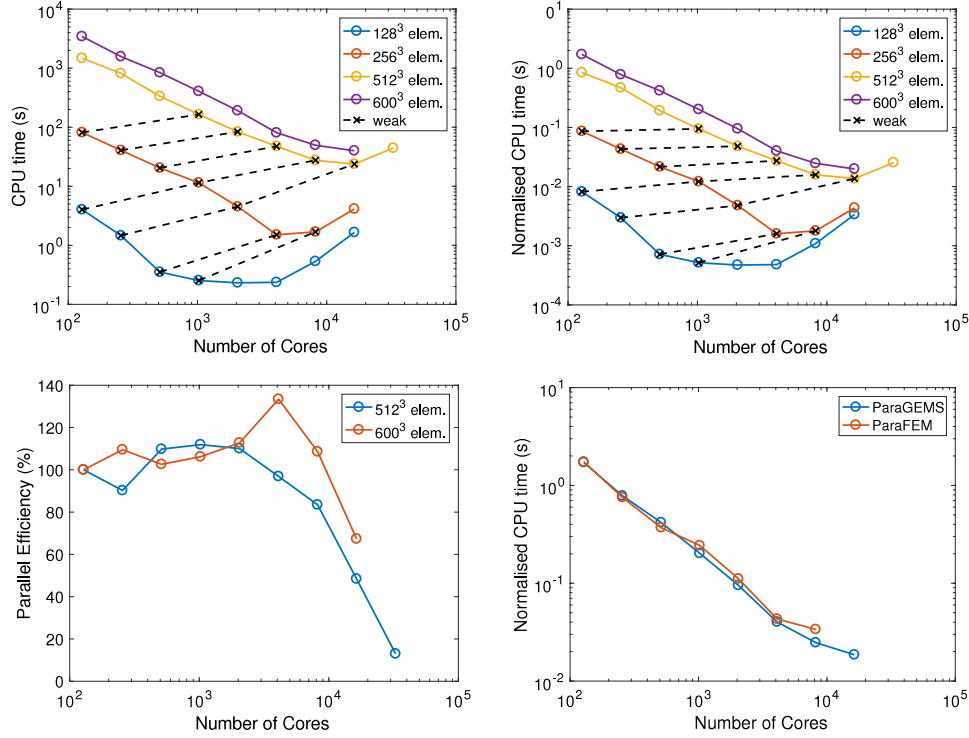
Fig. 4. Strong and weak scaling of pg123 solver using raw CPU time (top left); CPU time normalised by solver iterations (top right); parallel efficiency (bottom left); and compared to p123 (ParaFEM-FE) normalised by solver iterations (bottom right).

that components parallel to the primal edge remain unchanged by the mapping sequence flat-sharp. As a result, this new mapping makes the alternate decompositions of the constitutive relation for linear elasticity presented in Ref. [2] become mathematically identical.

3.3. Heterogeneous properties and evolving topologies

Relative to continuum approaches like FE, it is straight forward with DEC to both introduce discrete heterogeneity cell-by-cell and change the topology of the mesh. Fig. 5 shows the solution of the Laplace equation (left) and its gradient (right) with a random diffusion coefficient in each cell. This is shown for both the initial mesh (top) and after new discontinuities are introduced (bottom). This is a small sample of the types of simulations made possible with the integrated libraries.

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 in computer graphics [19]. 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 of new and evolving discontinuities with minimal modification of system matrices [8]. In FE, for example, the entire system matrix needs to be reformed to account for a 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 DEC into ParaFEM, many of the functions can also be adapted to implement

other discrete forms of exterior calculus, such as recent work of Barbetov et al. [20] based on Forman's combinatorial differential forms [21,22]. Therefore, the integration presented in this paper may have impact in broader theoretical research.

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 integration of ParaGEMS is expected to broaden and extend these opportunities. For example, the outputs will support new innovations promised by the authors in the UK Collaborative Computational Project: CCP-WSI+ (wave–structure interaction plus) which is focused on advancing offshore energy generation. Assessing the evolving structural performance of offshore structures is vitally important as they must be lightweight to facilitate shipping and installation, while also withstanding constant wave and wind action, seasonal heat cycles, and corrosion in-service.

Finally, this work can be used as a template for integrating DEC into other scalable libraries, such as FEniCS [23], Firedrake [24] or deal.II [25], and their associated workflows. This could lead to broad impact across many different fields.

5. Conclusions

Discrete Exterior Calculus (DEC) is a promising new approach for modelling the proprieties of materials according to their discrete microstructure at the mesoscale, especially those with heterogeneities and emerging or evolving discontinuities. To support continued development and application of DEC in solid mechanics, this paper presents a high-performance computational (HPC) research platform created by integrating the DEC library ParaGEMS into the well-established parallel finite element (FE) library ParaFEM. Existing ParaFEM miniApps for scalar diffusion
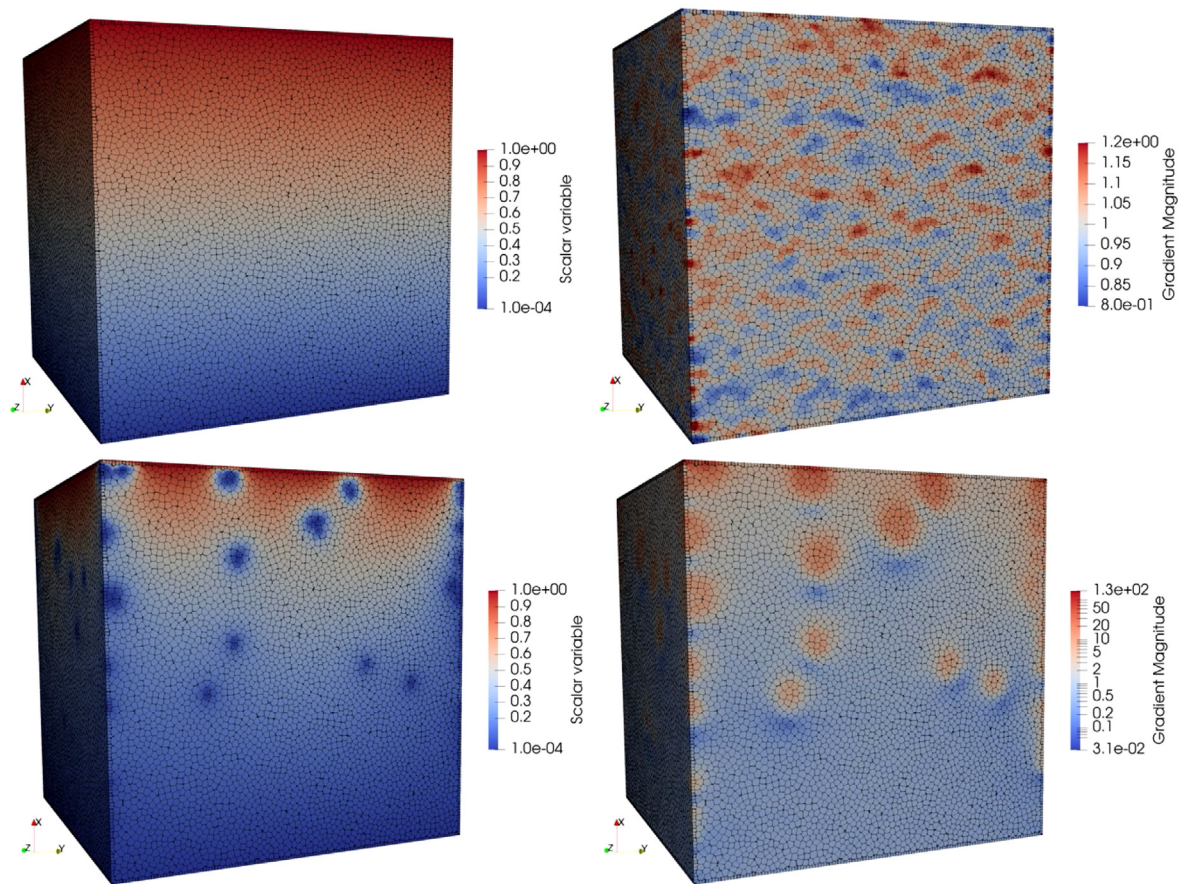
Fig. 5. Solution of Laplace equation (left) and gradient (right) for material with discretely heterogeneous diffusion coefficient before (top) and after (bottom) the introduction of discontinuities. (Note: the linear and log axes for the gradient).

and linear elasticity are adapted by simply replacing the individual contributions from FE elements with DEC formalism. FE elements are first independently decomposed into micro-complexes of one or more simplices and processed using local information. A new and local discrete musical isomorphisms is implemented to support the formulation of linear elasticity within this framework. Geometric information of the Dual Voronoi cells partitioned across multiple micro-complexes are finally superimposed, but only during assembly of the global system matrix. The integrated libraries make use of the same IO, communication patterns, partitioning and scalable solver native to ParaFEM. The DEC miniApps presented in this article are all variations on the Laplace equation, which achieved 80% strong parallel efficiency on ARCHER2 with over 8000 cores. Furthermore, trends indicate that this may extend to higher core counts with larger meshes. Finally, simulations of discretely heterogeneous and discontinuous materials are also presented, highlighting the benefits of DEC-based simulation enabled by the framework.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Pieter Boom, Andrey Jivkov, Lee Margetts reports financial support was provided by Engineering and Physical Sciences Research Council. Pieter Boom, Andrey Jivkov, Lee Margetts reports financial support was provided by ARCHER2.

Data availability

All software and data used in this article is available on Github.

Acknowledgements

References

[1] Leake J, Holmes D, Bridges A, Barber B, Best C, Hudson D. 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] Boom PD, Kosmas O, Margetts L, Jivkov AP. A geometric formulation of linear elasticity based on discrete exterior calculus. Int J Solids Struct 2022;111345.

[3] Hirani A. Discrete exterior calculus (Ph.D. thesis), California Institute of Technology; 2003.

[4] Nakshatrala KB, Hirani AN, Chaudhry JH. Numerical method for darcy flow derived using dis- crete exterior calculus. Int J Comput Methods Eng Sci Mech 2015;16:151–69.

[5] Mohamed M, Hirani A, Samtaney R. Discrete exterior calculus discretization of incompressible Navier–Stokes equations over surface simplicial meshes. J Comput Phys 2016;312:175–91.

[6] Chen S, Chew W. Discrete electromagnetic theory with exterior calculus. In: 2016 Progress in electromagnetic research symposium. 2016, p. 896–7.

[7] Yavari A. On geometric discretization of elasticity. J Math Phys 2008;49(2):022901.

[8] Boom PD, Seepujak A, Kosmas O, Margetts L, Jivkov AP. Parallelized discrete exterior calculus for three-dimensional elliptic problems. Comput Phys Comm 2022;279:108456.

[9] Smith IM, Griffiths DV, Margetts L. Programming the finite element method. 5th ed.. John Wiley & Sons, Ltd; 2014.

[10] Hewitt S, Margetts L, Shterenlikht A, Revell A. A massively parallel multiscale CAFE framework for the modelling of fracture in heterogeneous materials under dynamic loading. Adv Eng Softw 2020;139:102737.

[11] Cebamanos L, Shterenlikht A, Margetts L, Arregui-Mena JD. Open source exascale multi-scale framework for the UK solid mechanics community ARCHER eCSE05-05 report. 2017, http://www.archer.ac.uk/community/eCSE/eCSE05-05/ecse0505-techreport.pdf.

[12] Evans L, Margetts L, Casalegno V, Leonard F, Lowe T, Lee P, et al. Thermal characterisation of ceramic/metal joining techniques for fusion applications using X-ray tomography. Fusion Eng Des 2014;89(6):826–36.

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

[14] Hewitt S, Margetts L, Revell A, Pankaj P, Levrero-Florencio F. OpenFPCI: A parallel fluid–structure interaction framework. Comput Phys Comm 2019;244:469–82.

[15] Evans L, Mena J, Mummery P, Akers R, Surrey E, Shterenlikht A, et al. Use of massively parallel computing to improve modelling accuracy within the nuclear sector. Int J Multiphys 2016;10(2):215–36.

[16] Arregui-Mena JD, Margetts L, Griffiths D, Lever L, Hall G, Mummery PM. Spatial variability in the coefficient of thermal expansion induces pre-service stresses in computer models of virgin gilsocarbon bricks. J Nucl Mater 2015;465:793–804.

[17] Levrero-Florencio F, Manda K, Margetts L, Pankaj P. Nonlinear homogenisation of trabecular bone: Effect of solid phase constitutive model. Proc Inst Mech Eng H J Eng Med 2017;231(5):405–14.

[18] Falkingham PL, Bates KT, Margetts L, Manning PL. Simulating sauropod manus-only trackway formation using finite-element analysis. R Soc Palaeontol 2011;7(1):142–5.

[19] Elcott S, Tong Y, Kanso E, Schroder P, Desbrun M. Stable, circulation-preserving, simplicial fluids. ACM Trans Graph 2007;26(1). 4–es.

[20] Berbatov K, Boom PD, Hazel AL, Jivkov AP. Diffusion in multi-dimensional solids using Forman's combinatorial differential forms. Appl Math Model 2022;110:172–92.

[21] Forman R. Morse theory for cell complexes. Adv Math 1998;134:90–145.

[22] Forman R. Combinatorial Novikov-Morse theory. Internat J Math 2002;13(4):333–68.

[23] Fenics project. 2021, [Online]. Available: https://fenicsproject.org. [Accessed 21 Septmeber 2022].

[24] Firedrake project. 2022, [Online]. Available: https://www.firedrakeproject.org. [Accessed 21 September 2022].

[25] deal.II. 2022, [Online]. Available: https://www.dealii.org. [Accessed 21 September 2022].