Exascale Vision for Geometry

Joaquim Peiró, David Moxey, Spencer Sherwin, James Slaughter
Imperial College London and University of Exeter

With contributions from Bob Haimes (MIT), John Chawner (Cadence/Pointwise) & Henry Bucklow (ITI)

Current State of the Art

- **Spline-based techniques**
  - Boundary representation (BRep) based on combination of B-spline/NURBS faces, edges, points.
    - Faces are typically tensor-product based, i.e. blending of 3/4 discrete edges.
- **Discrete techniques**
  - Faceted abstraction e.g. STL.
    - Generally, a triangular surface mesh represented by surface nodes and cell normal.
    - Can be extended to high-order (e.g. Bézier curves, spherigons).
- **Spatial occupancy**
  - Pixel/voxel approach
    - Takes a regular Cartesian/octree grid, combining this with metadata that states whether a given cell lies inside or outside computational domain.

ELEMENT

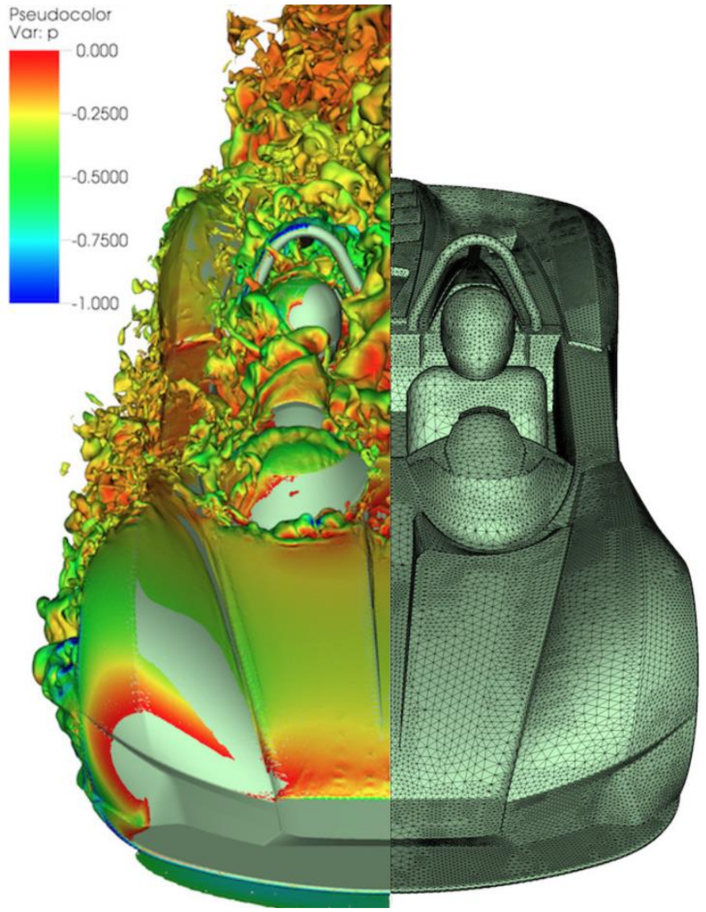
Workshop Learnings – Exascale Bottlenecks

- *Retention of CAD definition through the solution process to allow for adaptivity and optimisation.*
- *Lack of distribution methodology for current CAD definitions.*
- *Poor adherence to current industrial standards or use of proprietary formats making creation of a “one-size-fits-all-approach" almost impossible.*

ELEMENT

Geometry Retention

- ***Current techniques and approaches do not tend to retain geometry information through the simulation pipeline.***
  - Presents issues for MDAO simulations (e.g. fluid-structure interaction), or for parametric & optimisation problems where the geometry may need to be queried and altered in-situ.
  - Similarly, adaptive simulations (likely a major requirement at exascale) will need access to CAD in a similar manner.
  - Some APIs exist, e.g. OpenCascade (OCC), CADFix API – but may still contain implementation challenges:
    - libraries often large (e.g. thousands of classes in OCC) and targeted more at design than for running in-situ;
    - most software in this area is commercial;
    - rarely parallelized and sometimes not even multi-threaded; makes embedding in hybrid MPI/OpenMP codes potentially challenging.



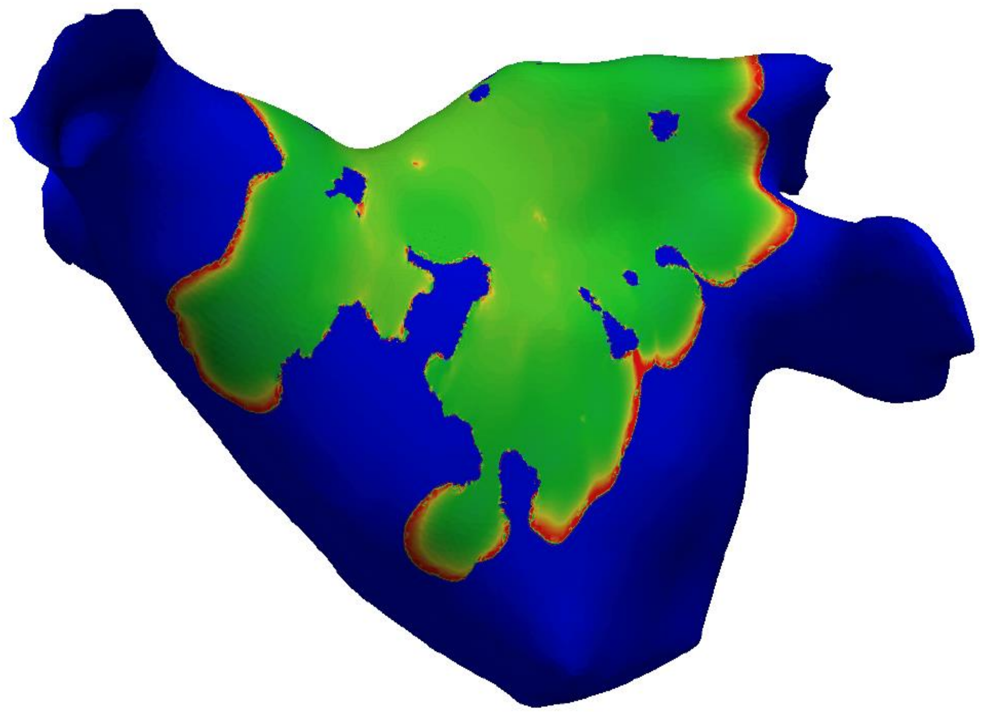
Geometry Portability

- ***Traditional geometry links to solvers are not designed to be parallelised.***
  - Only support sequential execution for build and query operations.
  - Kernels often external to simulation pipeline and poorly integrated – due to design decision or as they are proprietary.
  - Some research has been done in this area – for example, EGADslite (MIT) is designed specifically for distributed CAD.
  - CAD storage at node level could reduce need for partitioning in the interim.

Geometry Standards

- ***Geometry definition differs between disciplines.***
  - Most commercial software favours Mechanical CAD (MCAD) definitions.

- ***Simulation intent drives model design decisions.***
  - Lack of universality between simulation disciplines.
    - e.g. trailing edges of aerofoils for prism layers – not necessarily required for structural-type simulations.
    - Multiple definition handling might be a necessity for exascale.
  - Each discipline has its own spatial and feature refinement requirements.
    - 'Physics-Intelligent clean-up tools'.



Short Term Research Agenda

- ***Clean-up and repair tools.***
  - Greater size and complexity means that robustness is still a challenge, and emphasis is needed to ensure initial base meshes are fit for purpose without human intervention.
  - Therefore, we need tools to ensure simulation topologies are clean and robust.
    - Process still mostly manual and extremely time consuming – as most simulation geometry definitions are abstracted from the MCAD definition (e.g. fastener removal).
  - However, scaling up to extremely complex CAD geometries is likely to be extremely difficult.

ELEMENT

Medium Term Research Agenda

- ***Development and integration of CAD query APIs.***
  - Development of lightweight CAD engines that:
    - present a minimum-level API that is easier for solver integration;
    - consider use of lighter-weight CAD representations;
    - should perhaps aim to minimize prerequisites for easier use in HPC environments;
    - have greater emphasis on reducing computational overheads, e.g. in memory footprint.
  - These could initially focus on serial execution, with extension to allow for parallel/distributed geometry queries.

Long Term Research Agenda

- ***CAD partitioning and portability***
  - CAD partitioning could present a major bottleneck for exascale computation.
    - Shorter term solution could be storage at node-level, although this has associated memory and communication issues.
  - Reduced approximate form of the CAD.
    - Possibly a high-order discrete representation that can be distributed more readily (although this introduces similar load balancing concerns).
    - Kernels that allow for dynamic topological changes in the approximation (as needed for dynamic problems).
- ***Unified Agreement on CAD Standards for Portability***
  - Unlikely to ever occur – most definitions proprietary, although would greatly assist with research and development in this space!
  - Perhaps an alternative & more achievable target would be to produce a new standard for reduced/lightweight representations at exascale.

ELEMENT