Computational cost of simplices versus parallelotopes for Galerkin methods

A. Angeloski[1], X. Roca[2], J. Peraire[2], and A. Huerta[1]

[1] *Universitat Politècnica de Catalunya - BarcelonaTech, Barcelona, 08034, Spain*
[2] *Massachusetts Institute of Technology, Cambridge, MA 02139, USA*

1 Introduction

A comparison of the computational cost associated with *parallelotopes* (quadrilaterals and hexahedra) and *simplices* (triangles and tetrahedra) is presented. Specifically, we compare the cost of element-wise and global operations for different Galerkin methods and interpolation degrees. The estimates show that simplices outperform parallelotopes for element-wise operations, except for linear elements. On the contrary, parallelotopes outperform simplices for global operations that depend non-linearly on the number of mesh faces. For instance, solving a hybridized discontinuous Galerkin linear system with a sparse direct solver. Similar comparisons have been presented before but only for triangles and quadrilaterals [1, 2, 3].

Mesh hypothesis. We consider structured meshes of dimension $d = 2, 3$ and interpolation degree p, where the number of boundary faces is negligible compared with the number of interior faces. Given a structured quadrilateral (hexahedral) mesh, the triangular (tetrahedral) mesh is obtained by dividing each quadrilateral (hexahedron) in 2 triangles (6 tetrahedra), Figure 1. In the following, we denote the number of elements (faces) with n_e (n_f), and the number of degrees of freedom per element (face) with $\mathtt{ndof}_e$ ($\mathtt{ndof}_f$).

Proposed comparison. Let us consider an operation with the cost

$$\left(n_{entity}\right)^r \left(\mathtt{ndof}_{entity}\right)^q,$$

where $r, q \geq 1$, and *entity* can be either an element or a face. Then boxes are better than simplices if and only if

$$\left(\frac{n_{entity}^a}{n_{entity}^b}\right)^r \left(\frac{\mathtt{ndof}_{entity}^a}{\mathtt{ndof}_{entity}^b}\right)^q \geq 1,$$

where the superscripts a and b stand for simplices and parallelotopes, respectively. Assuming the same precision for parallelotopes and simplices, we
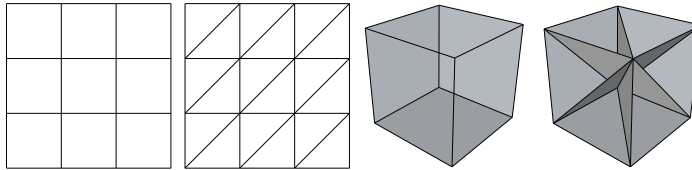
Fig. 1: Splitting quadrilaterals (hexahedra) in triangles (tetrahedra).

	n_e^a	n_f^a		Dim.	p	1	5	10	∞
	n_e^b	n_f^b	$\mathtt{ndof}_e^a$	2D		$\frac{3}{4}$	$\frac{7}{12}$	$\frac{6}{11}$	$\frac{1}{2}$
			$\mathtt{ndof}_e^b$	3D		$\frac{1}{2}$	$\frac{7}{27}$	$\frac{26}{121}$	$\frac{1}{6}$
2D	2	$\frac{3}{2}$							
3D	6	4	$\mathtt{ndof}_f^a$	2D		1	1	1	1
			$\mathtt{ndof}_f^b$	3D		$\frac{3}{4}$	$\frac{7}{12}$	$\frac{6}{11}$	$\frac{1}{2}$

Table 1: Ratios of: (left) number of entities, and (right) $\mathtt{ndof}$ per element (face).

present in Table 1a and Table 1b the corresponding ratios for the number of entities (n_{entity}) and the number of degrees of freedom per entity ($\mathtt{ndof}_{entity}$), respectively. Simplices have higher number of elements (faces), but less $\mathtt{ndof}$ per element (face). Thus, for element-wise operations that depend linearly with the number of elements, and non-linearly with the $\mathtt{ndof}$ per element, we can expect that simplices perform better. On the contrary, for global operations that depend non-linearly with the number of faces, parallelotopes should perform better due to the smaller number of faces.

In the remainder of this note, we compare the cost of element-wise and global operations for different interpolation degrees and Galerkin methods (Section 2). We consider the following methods: continuous Galerkin with and without static condensation (CG and CG(NSC)) [4], compact discontinuous Galerkin (CDG) [5], and the hybridizable discontinuous Galerkin method (HDG) [6, 7]. We finalize with the concluding remarks (Section 3).

2 Cost comparison: simplices versus parallelotopes

2.1 Element-wise operations

Creating element matrices. To compute an elemental matrix, we approximate with $\mathtt{ndof}_e$ integration points an integral that pairs $\mathtt{ndof}_e$ test functions with $\mathtt{ndof}_e$ trial functions. This results in a cost of $\mathtt{ndof}_e^3$ operations. Thus, the total cost for creating the element matrices is $E = n_e \cdot \mathtt{ndof}_e^3$. For straight-sided elements, the Jacobian is constant and therefore, the cost can be reduced to $E_J = n_e \cdot \mathtt{ndof}_e^2$.

Removing the inner degrees of freedom. For high-order methods is standard to parameterize the degrees of freedom in the interior of the elements in terms of the degrees of freedom on the faces. This results in a reduced global system that only depends on the degrees of freedom on the faces. This
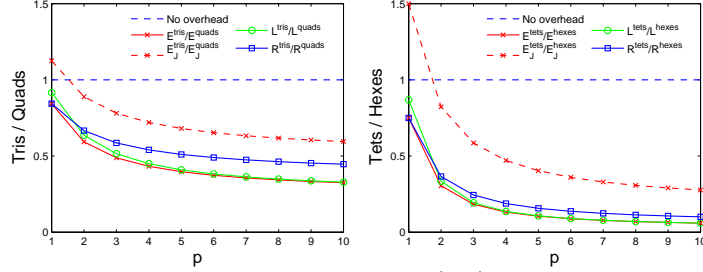
Fig. 2: Cost ratio for element-wise operations: (left) triangles over quadrilaterals, and (right) tetrahedra over hexahedra.

technique corresponds to the static condensation in CG and the solution of the local problem in HDG. For an element, the cost of this parameterization is dominated by the cost of inverting a dense elemental matrix, I_e. Thus, the total cost is

$$L = n_e \cdot I_e = n_e \cdot \left(\frac{8}{3} \texttt{ndof}_e^3 + \frac{9}{2} \texttt{ndof}_e^2 - \frac{7}{6} \texttt{ndof}_e \right). \tag{1}$$

Recovering the inner degrees of freedom. Once the global system for the degrees of freedom on the faces is solved, we can recover the inner degrees of freedom. To this end, we use the already computed parameterization in terms of the degrees of freedom on the faces. For one element this cost is dominated by the multiplication of dense matrices of size $\texttt{ndof}_e \times (d+1) \cdot \texttt{ndof}_f$ for simplices, and $\texttt{ndof}_e \times 2d \cdot \texttt{ndof}_f$ for parallelotopes, where d is the dimension. The resulting cost is

$$R = n_e \cdot R_e = n_e \cdot (2(d+1) \cdot \texttt{ndof}_f \cdot \texttt{ndof}_e^2)$$

for simplices, and

$$R = n_e \cdot R_e = n_e \cdot (4d \cdot \texttt{ndof}_f \cdot \texttt{ndof}_e^2)$$

for parallelotopes.

Results for element-wise operations. Figure 2 shows the ratios of the costs associated with triangles (tetrahedra) and quadrilaterals (hexahedra) for: constructing the element matrices with and without constant Jacobian (E_J and E), solving the local problem (L), and reconstructing the local variables (R). Note that all element-wise operations depend linearly with the number of entities, and non-linearly with the $\texttt{ndof}$ per entity. Hence, simplices are computationally cheaper for element-wise operations.

2.2 Solving the global problem

Pre-conditioned iterative solver. The cost per iteration is dominated by the cost of forming the Krylov subspace, which consists of performing
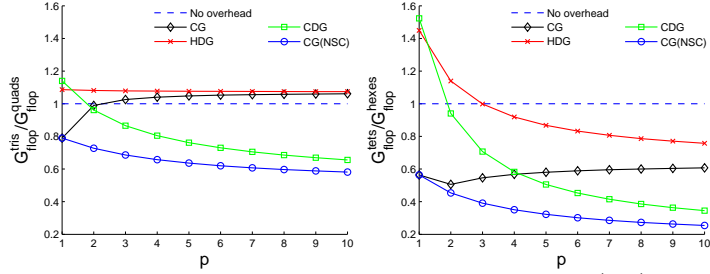
Fig. 3: Cost ratios for one iteration of an iterative solver: (left) triangles over quadrilaterals, and (right) tetrahedra over hexahedra.

the sparse matrix-vector product $y_k = Ax_k$, and applying the M^{-1} pre-conditioning operator. A reasonable pre-conditioner would be incomplete LU factorization with zero fill-in (ILU0) [8]. It can be shown that the cost per iteration for Galerkin methods when static condensation is used grows with $n_f \cdot \mathtt{ndof}_f^2$, and when static condensation is not used with $n_e \cdot \mathtt{ndof}_e^2$.

Figure 3 shows the results for different interpolation degrees (p) and methods (CG [4], CG(NSC)[4], CDG[5], and HDG[6]).

Solving the HDG global system with a sparse direct solver. The HDG method leads to a sparse matrix structured in dense blocks (of size $\mathtt{ndof}_f$). We assume that nested dissection [9, 10] is used to renumber the degrees of freedom on the edges (faces). Thus, the number of floating operations to solve the global system with a sparse direct solver is

$$G = n_f^{\frac{d+1}{2}} \cdot I_f = d^{\frac{d+1}{2}} \cdot n_e^{\frac{d+1}{2}} \cdot I_f, \tag{2}$$

for parallelotopes, and

$$G = n_f^{\frac{d+1}{2}} \cdot I_f = \left(\frac{d+1}{2}\right)^{\frac{d+1}{2}} \cdot n_e^{\frac{d+1}{2}} \cdot I_f, \tag{3}$$

for simplices. In these expressions, $n_f^{\frac{d+1}{2}}$ is the cost of factoring the system matrix, and $I_f = \left(\frac{8}{3}\mathtt{ndof}_f^3 + \frac{9}{2}\mathtt{ndof}_f^2 - \frac{7}{6}\mathtt{ndof}_f\right)$ is the cost of the inversion of a dense block of size $\mathtt{ndof}_f \times \mathtt{ndof}_f$. Figure 4 shows the cost ratios for solving the global system arising from HDG .

3 Concluding remarks

We conclude that simplices are more efficient than parallelotopes for element-wise operations, which, depend linearly on the number of elements and non-linearly on the $\mathtt{ndof}$ per element. This is also the case for the cost of one iteration of a preconditioned iterative solver, except in 2D, where quadrilaterals are more efficient than triangles for reduced global linear systems (such
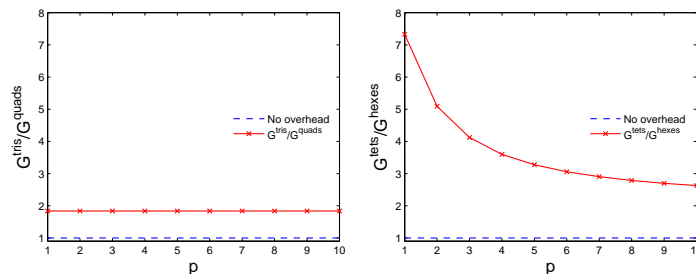
Fig. 4: Cost ratio for solving the HDG global system with a sparse direct solver: (left) triangles over quadrilaterals, and (right) tetrahedra over hexahedra .

as in CG and HDG). If the inner degrees of freedom are not removed (such as in CG(NSC) and CDG), triangles are cheaper. In 3D, tetrahedra are always faster per iteration. Finally, parallelotopes are more efficient than simplices for operations where the cost grows non-linearly with the number of faces. For instance, the cost of solving the HDG global system with a sparse direct solver.

References

1. R. Kirby, S. Sherwin, and B. Cockburn. To CG or to HDG: A comparative study. *J. of Sci. Computing*, pages 1–30, 2011. 10.1007/s10915-011-9501-7.
2. D. Wirasaet, S. Tanaka, E.J. Kubatko, J.J. Westerink, and C. Dawson. A performance comparison of nodal discontinuous galerkin methods on triangles and quadrilaterals. *Int. J. for Num. Methods in Fluids*, 64(10-12):1336–1362, 2010.
3. P. E. J. Vos, S. J. Sherwin, and R. M. Kirby. From h to p efficiently: Implementing finite and spectral/hp element methods to achieve optimal performance for low- and high-order discretisations. *J. of Comput. Physics*, 229(13):5161 – 5181, 2010.
4. P. G. Ciarlet. *The finite element method for elliptic problems*. North-Holland Publishing Co., Amsterdam, 1978.
5. J. Peraire and P.-O. Persson. The compact discontinuous galerkin (CDG) method for elliptic problems. *SIAM J. on Sci. Computing*, 30(4):25, 2007.
6. B. Cockburn, J. Gopalakrishnan, and R. Lazarov. Unified hybridization of discontinuous galerkin, mixed, and continuous galerkin methods for second order elliptic problems. *SIAM J. on Num. Analysis*, 47(2):1319–1365, 2009.
7. N. C. Nguyen, J. Peraire, and B. Cockburn. An implicit high-order hybridizable discontinuous galerkin method for linear convection-diffusion equations. *J. Comput. Phys.*, 228:3232–3254, May 2009.
8. P.-O. Persson and J. Peraire. Newton-GMRES Preconditioning for Discontinuous Galerkin Discretizations of the Navier-Stokes Equations. *SIAM J. Sci. Comput.*, 30(6):2709–2733, October 2008.
9. A. George. Nested dissection of a regular finite-element mesh. *SIAM J. on Num. Analysis*, 10(2):345–363, 1973.
10. I. Duff. *Analysis of sparse systems*. PhD thesis, University of Oxford, 1972.