From h to p efficiently: strategy selection for operator evaluation on hexahedral and tetrahedral elements

C.D. Cantwell[a,1], S.J. Sherwin[b], R.M. Kirby[c], P.H.J. Kelly[d]

[a]*Department of Mathematics, Imperial College London, London, UK*
[b]*Department of Aeronautics, Imperial College London, London, UK*
[c]*School of Computing, Univ. of Utah, Salt Lake City, Utah, USA*
[d]*Department of Computer Science, Imperial College London, London, UK*

Abstract

A spectral/hp element discretisation permits both geometric flexibility and beneficial convergence properties to be attained simultaneously. The choice of elemental polynomial order has a profound effect on the efficiency of different implementation strategies with their performance varying substantially for low- and high-order spectral/hp discretisations. We examine how careful selection of the strategy minimises computational cost across a range of polynomial orders in three dimensions and compare how different operators, and the choice of element shape, lead to different break-even points between the implementations. In three dimensions, higher expansion orders quickly lead to a large increase in the number of element-interior modes, particularly in hexahedral elements. For a typical boundary-interior modal decomposition, this can rapidly lead to a poor performance from a global approach, while a sum-factorisation technique, exploiting the tensor-product structure of elemental expansions, leads to more optimal performance. Furthermore, increased memory requirements may cause an implementation to show poor run-time performance on a given system, even if the strict operation count is minimal, due to detrimental caching effects and other machine-dependent factors.

Keywords: spectral h/p element, optimisation, code performance

1. Introduction

Spectral/hp element solvers are now widespread throughout the fluid dynamics community, as well as in many other areas of applied mathematics and engineering. Applications include incompressible fluid problems such as biomedical flows and flow control, turbulence models, structural mechanics, acoustics, electrophysiology, climate and geology modelling.

The benefits of using these high-order solvers stem from both the geometric flexibility offered by the elemental decomposition of the domain combined with the high accuracy and preferential convergence properties of a spectral method. Unlike linear finite-element or pure spectral techniques spectral/hp methods exhibit a broad scope for optimisation, not only through mesh refinement and polynomial order, but through careful choice of evaluation strategies for a given numerical operator. The importance of choosing the correct strategy in three dimensions should not be underestimated, as is evident from the results presented in Section 3. Even in two dimensions, a change of polynomial order by one may incur a runtime performance penalty on the order of 50%, if the evaluation strategy is not also adjusted to the optimal choice for the new discretisation.

In formulating a spectral/hp element method a domain is divided into a tessellation $\mathcal{K}$ of non-overlapping elements on which one or more solution variables are expanded in terms of polynomial functions up to a fixed order, P [1]. The typical choice for the polynomial functions are Legendre polynomials due to their orthogonality and favourable numerical properties [2]. The characteristics of these functions lie outside the scope of this paper. The typical choice for P varies between communities. Those from the finite-element community typically use expansions up to fourth-order [3], while those in the spectral/hp element community typically consider polynomial orders up to 15th order [2, 4]. For the purposes of this study we consider polynomial orders in the range of $1 \leq P \leq 10$, since this is sufficient for our analysis.

In constructing a spectral/hp expansion each elemental region is mapped onto a reference element on which the basic operations of integration and differentiation are defined. In two dimensions, these are typically quadrilaterals or triangles, but three dimensions encompasses a broader selection of hybrid shapes. These include hexahedrons, prisms, pyramids and tetrahedrons. Each of these regions exposes its own performance characteristics and while they offer great geometric flexibility, they may introduce complexities in choosing an optimal evaluation strategy. In this paper we will restrict ourselves to comparison of hexahedral and tetrahedral elements. Expansions in three dimensions are formed as a tensor product of one-dimensional polynomials. This permits the use of strategies which may dramatically reduce the operation count, and therefore improve the overall performance, of a given operator when compared with a naive local matrix implementation.

Email addresses: c.cantwell@imperial.ac.uk (C.D. Cantwell), s.sherwin@imperial.ac.uk (S.J. Sherwin), kirby@cs.utah.edu (R.M. Kirby), p.kelly@imperial.ac.uk (P.H.J. Kelly)
[1]Corresponding author

Local elemental modes are extended to a global context through direct stiffness assembly, typically with the enforcement of a C^0 continuity constraint. A sparse, invertible assembly matrix describes the scattering of global coefficients onto their corresponding local elemental coefficients allowing the problem, and therefore the evaluation of the operators, to be formulated in either a global or local context. This local-to-global mapping is used to construct a global, bandwidth-optimised, matrix system through which operations may be performed across all elements simultaneously. Conversely, elemental evaluation may be performed using a local matrix operation, or by exploiting the tensorial nature of the expansions and using a sum-factorisation approach [5]. The performance benefits of the latter have been noted in the literature [6], and the evaluation may be expressed using matrix-matrix multiplications which may be further optimised by the BLAS subsystem. The formulation of each of these strategies is detailed in Section 2.

In any spectral/hp implementation, the specifications of the system on which it is used, and the related software libraries, will affect its performance. The wall time taken to solve a particular problem will be affected by factors beyond the theoretical operation count, with the characteristics of the hardware and external libraries (such as BLAS and LAPACK) at a given problem size having a non-negligible effect. This paper will therefore focus on discussing the run-time performance of the various strategies to present a picture of the real performance of such operations in a three-dimensional context.

2. Spectral h/p element discretisation

We first summarise our spectral/hp element formulation for hexahedral and tetrahedral elements in reference-space. A more detailed construction may be found in Karniadakis and Sherwin [2]. We use a modified form of the Legendre basis in which the interior modes are zero on the boundary through multiplication by linear factors, while still maintaining the numerical efficiencies of the expansion. This allows for greater numerical optimisation of global strategies through boundary-interior decomposition.

2.1. Reference space expansions

For the hexahedral region we extend the modified set of one-dimensional Legendre polynomials, $\{\psi_p(\xi)\}$, to form a three-dimensional basis through a tensorial construction. The standard hexahedral reference region is $Q^3 = \left\{(\xi_1, \xi_2, \xi_3) \in [-1, 1]^3\right\}$ on which the basis functions take the form

$$\phi_n(\xi_1, \xi_2, \xi_3) = \psi_p(\xi_1)\psi_q(\xi_2)\psi_r(\xi_3).$$

This defines the standard hexahedral elemental expansion, $\Omega_{st}(Q^3)$. A solution defined on this region may be expanded in terms of these functions as

$$\begin{aligned} u(\xi_1, \xi_2, \xi_3) &= \sum_{n\in\mathcal{N}} \phi_n(\xi_1, \xi_2, \xi_3)\hat{u}_n \\ &= \sum_{p=0}^{P}\sum_{q=0}^{P}\sum_{r=0}^{P} \psi_p(\xi_1)\psi_q(\xi_2)\psi_r(\xi_3)\hat{u}_{pqr} \end{aligned} \quad (1)$$

where $\hat{\mathbf{u}}$ is the coefficient space representation. Note that in the hexahedral expansion the ψ_p, ψ_q and ψ_r are independent of each other.

The tetrahedral region, when defined in terms of orthogonal Cartesian coordinates, is $\mathcal{T}^3 = \{-1 \le \xi_1, \xi_2, \xi_3, \xi_1 + \xi_2 + \xi_3 \le -1\}$. This region does not have the constant limits necessary to explot the tensorial expansion set out in the hexahedral case. To fit the tetrahedron into this framework we employ a coordinate transform[7] from the Cartesian coordinate system (ξ_1, ξ_2, ξ_3) onto a non-orthogonal coordinate system (η_1, η_2, η_3). For the triangular expansion [8], such a mapping is

$$\begin{aligned} \eta_1 &= 2\frac{1+\xi_1}{1-\xi_2} - 1 \\ \eta_2 &= \xi_2 \end{aligned}$$

Repeated application of this mapping to the orthogonal coordinate system in three dimensions leads to a local mapping for tetrahedra [9],

$$\begin{aligned} \eta_1 &= \frac{2(1+\xi_1)}{-\xi_2 - \xi_3} - 1, \\ \eta_2 &= \frac{2(1+\xi_2)}{1-\xi_3} - 1, \\ \eta_3 &= \xi_3. \end{aligned}$$

Under this collapsed coordinate system, the tetrahedral region, $\mathcal{T}^3 = \{-1 \le \eta_1, \eta_2, \eta_3 \le 1\}$, is bounded by constant limits allowing the tensorial basis construction to be used. The standard tetrahedral expansion $\Omega_{st}(\mathcal{T}^3)$ is defined as

$$\phi_n(\xi_1, \xi_2, \xi_3) = \sum_{n\in\mathcal{N}} \psi_p(\eta_1)\psi_{pq}(\eta_2)\psi_{pqr}(\eta_3).$$

A consequence of the coordinate transform is the creation of two degenerate vertices which requires careful handling when generating meshes to ensure alignment of the collapsed coordinates. Furthermore, to maintain numerical efficiency, the one-dimensional expansion basis in the second direction varies with p, and similarly the corresponding basis in the third direction varies with both p and q. This leads to the following solution expansion

$$\begin{aligned} u(\xi_1, \xi_2, \xi_3) &= \sum_{n\in\mathcal{N}} \phi_n(\xi_1, \xi_2, \xi_3)\hat{u}_n, \\ &= \sum_{p=0}^{P}\sum_{q=0}^{P}\sum_{r=0}^{P} \psi_p(\eta_1)\psi_{pq}(\eta_2)\psi_{pqr}(\eta_3)\hat{u}_{pqr}. \end{aligned} \quad (2)$$

2.2. Local and global expansions

To represent a solution on an arbitrary local elemental region, Ω^e we construct a bijective linear mapping, $\chi_i^e : \Omega_{st} \rightarrow \Omega_e$. We assume the domain consists entirely of non-deformed elements and therefore leads to a map with constant positive Jacobian on each element,

$$\begin{aligned} x_1 &= \chi_1^e(\xi_1, \xi_2, \xi_3), \\ x_2 &= \chi_2^e(\xi_1, \xi_2, \xi_3), \\ x_3 &= \chi_3^e(\xi_1, \xi_2, \xi_3). \end{aligned}$$

If the faces of the elemental regions were instead defined by non-linear iso-parametric functions, the Jacobian would be dependent on the quadrature point.

A continuous Galerkin formulation dictates a degree of connectivity between the individual elements, typically in the form of a C^0 continuity condition. The boundary/interior decomposition of the elemental modes simplifies this construction since all interior elemental modes are themselves globally orthogonal. Therefore, given our tessellation $\mathcal{K}$ of elements, each with $\mathcal{N}$ elemental modes, the solution on the entire domain can be represented as

$$\begin{aligned} u(x_1, x_2) &= \sum_{m \in \mathcal{N}^g} \Phi_m(x_1, x_2) \hat{u}_m^g, \\ &= \sum_{e \in \mathcal{K}} \sum_{n \in \mathcal{N}} \phi_n^e(x_1, x_2) \hat{u}_n^e, \end{aligned}$$

where Φ_m are the $\mathcal{N}^g$ global modes. The mapping of local degrees of freedom to global degrees of freedom may be succinctly expressed as a highly sparse invertible local-to-global assembly matrix, $\mathcal{A}$,

$$\hat{\boldsymbol{u}}^l = \mathcal{A}\hat{\boldsymbol{u}}^g.$$

2.3. Evaluation strategies

The structured nature of the spectral/hp formulation allows for several approaches to evaluating numerical operators. In particular we will be concerned with the relative performance of computing a backward transform from coefficient space to physical space, an inner product, as well as the evaluation of mass and Helmholtz operators. We consider evaluating these operations using a global matrix operation, a sequence of local elemental matrix operations or through exploitation of the tensorial basis using sum-factorisation.

In the global context a sparse $\mathcal{N}^g \times \mathcal{N}^g$ matrix is constructed which directly solves for the global coefficients. This approach is typical of finite element formulations in which all modes are essentially elemental boundary modes and results in a significantly lower operation count than handling each element individually. At higher orders, the global matrix rapidly becomes very large, although substructuring techniques [10] can be used to dramatically improve the efficiency of this approach.

The remaining two evaluation strategies are performed at the elemental level. The global coefficients $\hat{\boldsymbol{u}}^g$ are scattered onto their corresponding local coefficients $\hat{\boldsymbol{u}}^l$ with which the operation is evaluated elementally as

$$\hat{y}_m^e = \sum_{n \in \mathcal{N}} a_e(\phi_m^e, \phi_m^e) \hat{u}_n^e \quad \forall (m, e) \in (\mathcal{N}, \mathcal{K}).$$

Here we use $a_e(v, u)$ to represent a general bi-linear operator typical of a weak Galerkin formulation of a PDE. The resulting vector, $\hat{\boldsymbol{y}}$ is then reassembled to give the global solution.

The sum-factorisation strategy[5] exploits the tensorial nature of the elemental basis. For simplicity we will consider just the backward transform in the hexahedral region in detail. The other operators may be expressed in a similar form. The expansion basis defined in Equation (1) can be reorganised as follows

$$u(\xi_{1i}, \xi_{2j}, \xi_{3k}) = \sum_{p=0}^{P} \psi_p(\xi_{1i}) \left\{ \sum_{q=0}^{P} \psi_q(\xi_{2j}) \left\{ \sum_{r=0}^{P} \hat{u}_{pqr} \psi_r(\xi_{3k}) \right\} \right\}. \tag{3}$$

In hybrid regions, the inter-dependence of the one-dimensional modes in the tensorial construction leads to a restriction on the ordering of the factorisation of Equ. (2). However, this does not prevent the same technique being applied to these regions.

Further to the reduction in operation count, the summation can be expressed as a sequence of matrix-matrix multiplications. To see this we define the order of the elemental modes $\hat{u}_{pqr}$ in the vector $\hat{\boldsymbol{u}}$ to be such that p (rows) runs fastest, followed by q (columns), and r (stacks) runs slowest. Let $\hat{\boldsymbol{U}}_{[P]}$ define a $P \times P^2$ matrix where each column corresponds to a *row* of coefficients in $\hat{\boldsymbol{u}}$; that is $\hat{\boldsymbol{u}}[iP : (i+1)P - 1]$, $0 < i < P^2$. The backward-transform operator from Equation (3) can now be expressed as a sequence of matrix-matrix multiplications,

$$\boldsymbol{U}_{[P]} = \left[\left(\hat{\boldsymbol{U}}_{[P]}^\top \boldsymbol{B}_{\boldsymbol{0}}^\top \right)^\top \boldsymbol{B}_1^\top \right]^\top \boldsymbol{B}_2^\top = \boldsymbol{B}_1 (\hat{\boldsymbol{U}}_{[P]}^\top \boldsymbol{B}_0^\top) \boldsymbol{B}_2^\top.$$

The other operators may be expressed in a similar way. For example, the inner product operator and mass-matrix operators become

$$\hat{\boldsymbol{U}}_{[\boldsymbol{P}]} = \left[\boldsymbol{B}_0^\top w(\boldsymbol{U}_{[Q]}) \boldsymbol{B}_1 \right]^\top \boldsymbol{B}_2,$$

and

$$\hat{\boldsymbol{U}}_{[\boldsymbol{P}]} = \left[\boldsymbol{B}_0^\top w \left(\boldsymbol{B}_1 \left[\hat{\boldsymbol{U}}_{[P]}^\top \boldsymbol{B}_0^\top \right] \boldsymbol{B}_2^\top \right) \boldsymbol{B}_1 \right]^\top \boldsymbol{B}_2,$$

respectively, where $w(\boldsymbol{U}_{[P]})$ applies the quadrature metrics to the coefficients.

2.4. Implementation and test system

The specific spectral/hp implementation used is Nektar++ [2], written in C++. The matrix-matrix and matrix-vector linear algebra operations are performed using the reference BLAS and LAPACK implementations available on the test system (Mac Pro with two 2.26Ghz 4-core processors, 2MB L2 cache, 8MB L3 cache, 16GB RAM) using `dgemm` and `dgemv`, respectively.

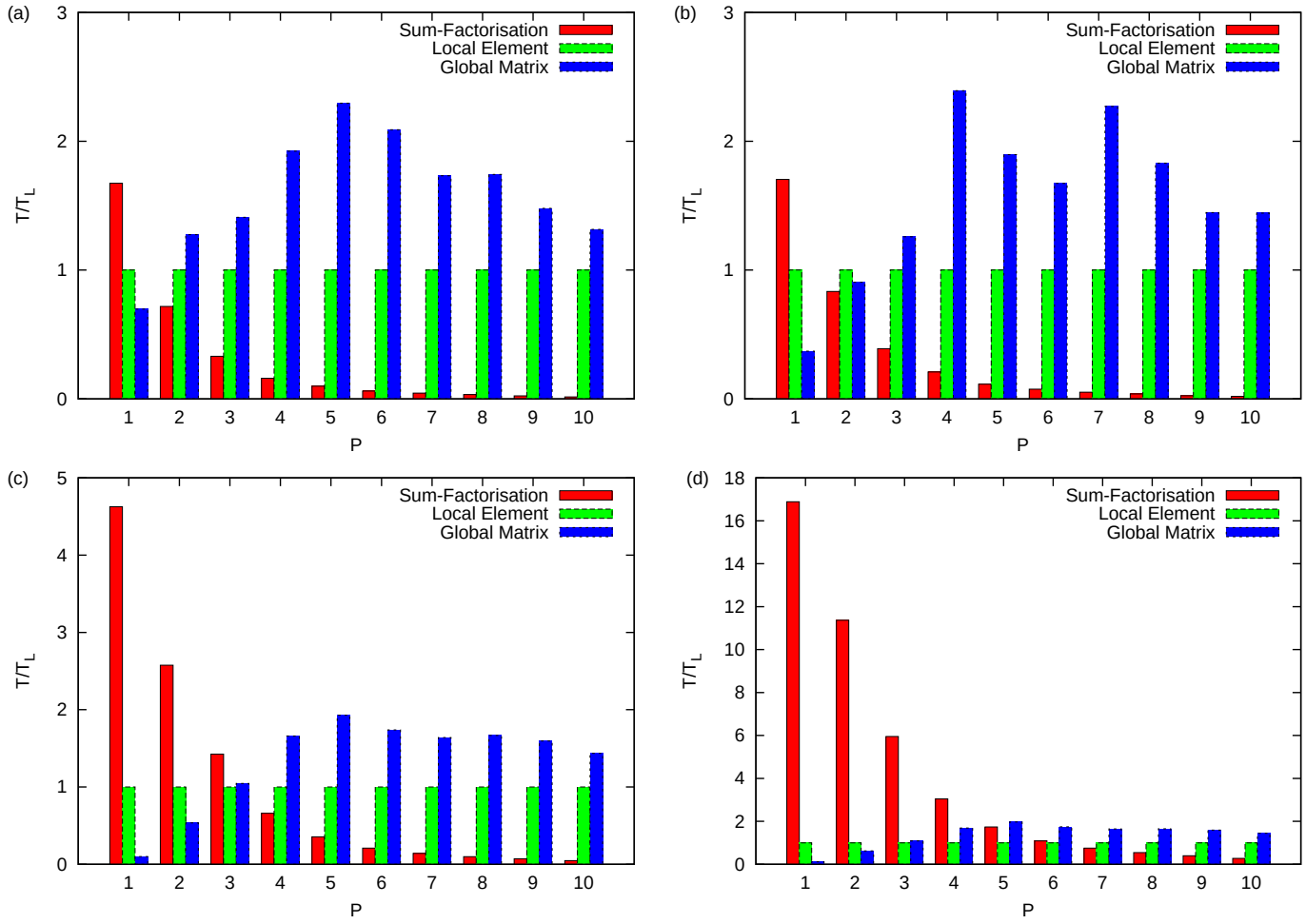

Figure 1: Comparative performance of (a) backward transform, (b) inner product, (c) mass matrix and (d) Helmholtz operators on a mesh of 64 hexahedral elements. All results are normalised by the local elemental performance for comparison. The break-even points apparent in this figure are largely independent of the choice of h and so this result is representative of larger numbers of elements.

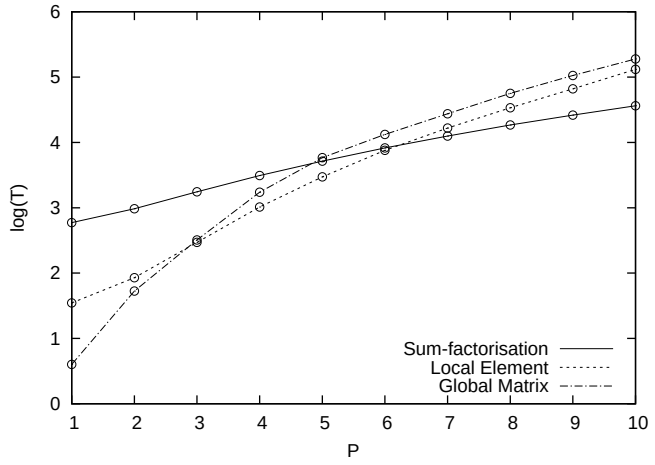

Figure 2: Absolute comparison of runtimes for the Helmholtz operator using 64 hexahedral elements. The number of microseconds required to evaluate the Helmholtz operator once is shown.

P	Global	Local	Sum-Fac.
Backward Transform	1	-	2-
Inner Product	1	-	2-
Mass Matrix	1-2	3	4-
Helmholtz Matrix	1-2	3-6	7-

Table 1: Table of optimal strategy selection for different operators on hexahedral meshes up to $P = 10$.

3. Results

Figure 1 shows the runtime performance of hexahedral elements for the three strategies and four numerical operators. We make a comparison of the strategies relative to the intermediate local elemental matrix strategy for clarity of comparison. In this figure, the results are computed using a cube mesh of 64 hexahedral elements, although the optimal strategies and break-even points do not significantly change with variation in h. We

[2] www.nektar.info

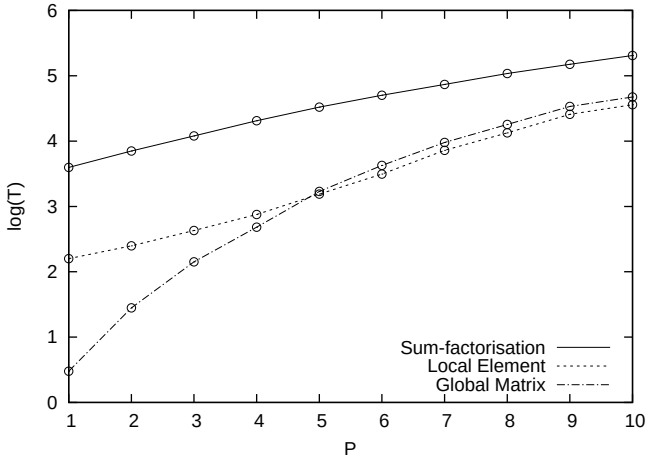

Figure 3: Absolute comparison of runtimes for the Helmholtz operator using 384 tetrahedral elements. The number of microseconds required to evaluate the Helmholtz operator once is shown.

P	Global	Local	Sum-Fac.
Backward Transform	1-2	3-4	5-
Inner Product	1-3	-	4-
Mass Matrix	1-4	5-10	-
Helmholtz Matrix	1-4	5-10	-

Table 2: Table of optimal strategy selection for different operators on tetrahedral meshes up to $P = 10$.

summarise the strategy break-even points for hexahedrons in Table 1.

An immediate observation is that the global matrix approach is only optimal at low-orders, typically order 1 or 2 polynomials. This in itself is not surprising as the operation count is far lower than for an elemental approach. However, the dominance of elemental boundary modes in three-dimensional elements continues to much higher orders, suggesting a global strategy may still have a lower operation count. At high orders it becomes rapidly sub-optimal, although it is surprising this approach does not provide better performance at orders as low as three. Figure 2 shows an absolute comparison of runtimes — measured in microseconds — for the three strategies when applied to the Helmholtz operator. Interestingly, the global runtime saturates at a rate not much higher than that of the local elemental strategy. This can be attributed to the performance gain provided by the multi-level static condensation (substructuring) employed in the global matrix implementation.

The performance of the sum-factorisation strategy is poor at low orders, particularly for the more complex mass and Helmholtz operators. In such cases the performance difference in relation to the global strategy could be as high as two orders of magnitude. The local elemental approach is typically only optimal for low to intermediate orders when performing complex operations such as those involving differentiation.

Figure 4 demonstrates a slight shift away from local strategies for tetrahedral meshes, with a global matrix approach giving the best performance up to fourth order for some operators. Sum-factorisation is particularly poor in this geometry when used with the mass and Helmholtz operators. At low orders, it can be up to three orders of magnitude slower than the global approach. This is due to the necessity of using a series of matrix-vector operations, rather than more optimised matrix-matrix operations, in the sum-factorisation to handle the interdependence of the basis modes in the second and third dimension. Consequently this eliminates the benefits of cache locality present in matrix-matrix operations. Again we summarise the strategy break-even points for tetrahedra in Table 2.

In comparing the performance of hexahedra and tetrahedra we observe the relative performance of the local matrix and global matrix approaches are quite similar for the backward transform and inner product. For more complex operations tetrahedral operations benefit from global strategies to a higher polynomial order, although for a large portion of the polynomial order spectrum, the local matrix approach is optimal.

4. Discussion

We have summarised the comparative performance of hexahedral and tetrahedral spectral/hp element discretisations for a range of polynomial orders commonly used in different communities. Operations within the spectral/hp formulation may be evaluated in either a global or local framework. Furthermore, in the local context, the tensorial construction of the elemental basis modes allows for the choice of sum-factorisation or elemental matrix evaluation. Ideally, a spectral/hp element code utilising a tensorial basis should support all three evaluation techniques to ensure a high performance over a broad range of polynomial orders. As with the two-dimensional case [11], the general principle is to use global strategies at low orders, local elemental strategies at intermediate orders, and sum-factorisation at high orders. This falls in line with the general approach taken by the various academic and industrial communities using low- and high-order finite element techniques. However, as these results demonstrate, there is no fixed rule applicable to all problems and the type of operation and elemental shapes involved are key factors in determining an optimal evaluation strategy for a given polynomial order on a particular system.

The dominance of elemental boundary modes in low-order expansions means a global approach should always offer the greatest performance. The implementation of the global strategy employs a multi-level static condensation technique which, through substructuring of the global degrees of freedom associated with elemental boundaries, allows a bandwidth-optimised matrix system to be produced. This system can be solved considerably faster than the original full or banded matrix system adding significant performance to this approach. At high polynomial orders the larger number of modes rapidly leads to large elemental matrices which are outperformed by a sum-factorisation approach for some operators. This is not surprising since the elemental operators are true three-dimensional operators which require $O(P^6)$ floating-point operations to apply, while the sum-factorisation approach consists of three matrix-matrix multiplies, each requiring just $O(P^4)$ floating-point op-

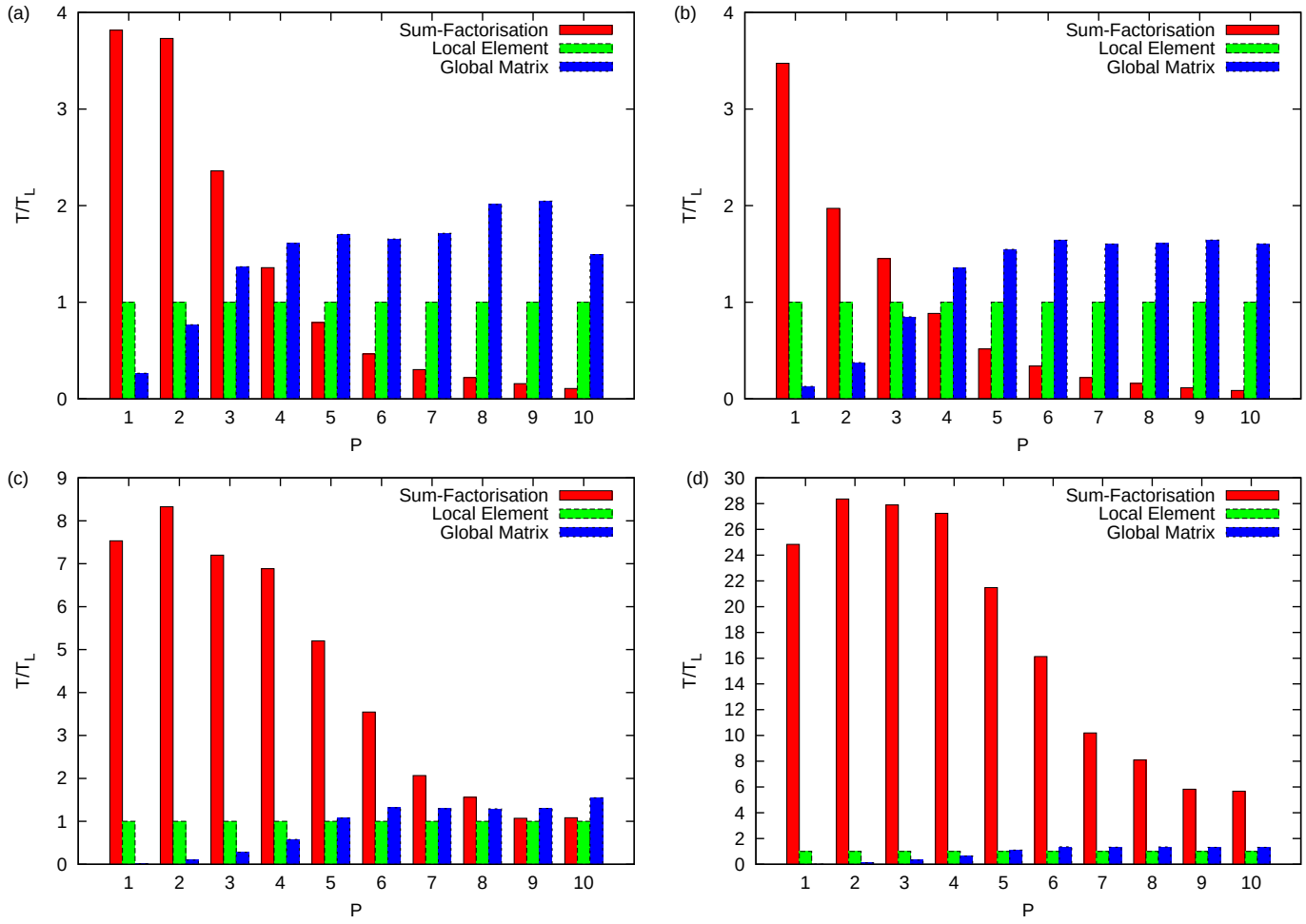

Figure 4: Comparative performance of (a) backward transform, (b) inner product, (c) mass matrix and (d) Helmholtz operators on a mesh of 384 tetrahedral elements.

erations. This is indeed the strength of the sum-factorisation approach.

The boundaries between the various strategies are not uniquely defined and they depend on the specification of the hardware and any performance enhancements offered by the operating system and linear algebra libraries. Certain systems support the implicit parallelisation of BLAS operations which can dramatically accelerate various large matrix operations and consequently shift the strategy boundaries by several polynomial orders.

We conclude the discussion by comparing these results with a similar study in the two-dimensional case [11]. In both a comparison of the quadrilateral and hexahedral regions, as well as the triangular and tetrahedral regions, we see a strong similarity between the relative runtimes across all four operators. There is a slight shift of the break-even points towards the lower end of the polynomial spectrum in the three-dimensional cases. This is attributable to the number of modes increasing as P^3 rather than P^2. The relative performance of the sum-factorisation at higher orders is consequently much greater in the three-dimensional case, while being especially poor at low orders.

5. Conclusions

This study has shown that the choice of strategy for the evaluation of operators in three dimensions is critical to attain the best performance from a spectral/hp element solver. Essentially, the differing performance of the various strategies is emphasised to a much greater extent in three dimensions and great care should be taken to select the best strategy for each operator on a given system.

Acknowledgements

The authors would like to acknowledge the valuable input of Peter Vos, Imperial College London.

References

1. Patera A. A spectral element method for fluid dynamics: laminar flow in a channel expansion. *J Comput Phys* 1984;**54**(3):468–488.
2. Karniadakis GE, Sherwin SJ. *Spectral/hp element methods for computational fluid dynamics*. Oxford: Oxford University Press; second edition ed.; 2005.
3. Hughes TJR. *The Finite Element Method*. New Jersey: Prentice-Hall; 1987.

4. Szabó B, Babuška I. *Finite Element Analysis*. New York: John Wiley & Sons; 1991.
5. Orszag SA. Spectral methods for problems in complex geometries. *Advances in computer methods for partial differential equations- III* 1979; :148–157.
6. Melenk JM, Gerdes K, Schwab C. Fully discrete hp-finite elements: fast quadrature. *Comp Meth Appl Mech & Engng* 2001;**190**(32-33):4339–4364.
7. Dubiner M. Spectral methods on triangles and other domains. *J Sci Comp* 1991;**6**(4):345–390.
8. Sherwin SJ, Karniadakis GE. A triangular spectral element method; applications to the incompressible navier-stokes equations. *Comp Meth Appl Mech & Engng* 1995;**123**(1-4):189–229.
9. Sherwin SJ, Karniadakis GE. Tetrahedral hp finite elements: Algorithms and flow simulations. *J Comput Phys* 1996;**124**:14–45.
10. Smith B, Bjorstad P, Gropp W. *Domain decomposition: parallel multilevel methods for elliptic partial differential equations*. Cambridge Univ Pr; 2004.
11. Vos PEJ, Sherwin SJ, Kirby M. From h to p efficiently: Implementing finite and spectral/hp element discretisations to achieve optimal performance at low and high order approximations. *J Comput Phys* 2010;**(in press)**.