Procedia Computer Science

Volume 51, 2015, Pages 1072–1081

ICCS 2015 International Conference On Computational Science

Comparison of the structure of equation systems and the GPU multifrontal solver for finite difference, collocation and finite element method

P. Lipski[1], M. Woźniak[1], and M. Paszyński[1]

AGH University of Science and Technology, Krakow, Poland

Abstract

The article is an in-depth comparison of numerical solvers and corresponding solution processes of the systems of algebraic equations resulting from finite difference, collocation, and finite element approximations. The paper considers recently developed isogeometric versions of the collocation and finite element methods, employing B-splines for the computations and ensuring C^{p-1} continuity on the borders of elements for the B-splines of the order p. For solving the systems, we use our GPU implementation of the state-of-the-art parallel multifrontal solver, which leverages modern GPU architectures and allows to reduce the complexity. We analyze the structures of linear equation systems resulting from each of the methods and how different matrix structures lead to different multifrontal solver elimination trees. The paper also considers the flows of multifrontal solver depending on the originally employed method.

Keywords: parallel computing, isogeometric analysis, collocation method, finite element method, multifrontal direct solvers, grammar-based solvers

1 Background and motivation

Essentially, each of the analyzed methods (finite difference method (FDM), collocation, finite element method (FEM)) yields a system of linear algebraic equations. The system is then passed to a solver — either sequential or parallel. The paper focuses on our GPU implementation of the state-of-the-art parallel multifrontal solver as described in [5, 6, 7, 8, 9], featuring logarithmic computational complexity with respect to the size of the matrix.

The paper compares the structure of the matrices generated by 1D FDM, collocation method and FEM and the corresponding forward elimination trees built within the solver. We have not found such a holistic comparison in the existing literature. We believe such a comparison may be of benefit to the researchers who deal with direct solvers and isogeometric analysis. We firstly derive the system of linear algebraic equations and multifrontal solver trees for FDM and FEM with hierarchical basis functions [3] and then switch to isogeometric bases.

 Selection and peer-review under responsibility of the Scientific Programme Committee of ICCS 2015

doi:10.1016/j.procs.2015.05.266

Isogeometric analysis (IGA) is recently developed approach [1] for computations that enables for more tightly integrating NURBS-based tools CAD with finite element analysis tools. The paper analyzes the isogeometric methods [2], i.e. isogeometric collocation method and isogeometric FEM [11], and deliver both matrices and elimination trees. The equation systems are significantly more complex in the case of two-dimensional problems. The paper also shows a few examples of the structure of such matrices.

In the final part of the paper we conclude with performance measures of a shared-memory implementation of the algorithm (see also [10, 12]) and useful remarks for the potential implementations of the solvers.

2 Finite difference method

Let us focus on a simple one-dimensional elliptic model problem with mixed Dirichlet and Neumann boundary conditions.

$$-\frac{d}{dx}\left(\frac{du\,(x)}{dx}\right) = 0 \tag{1}$$

$$u\,(0) = 0 \tag{2}$$

$$\frac{du\,(1)}{dx} = 1 \tag{3}$$

The global system of linear equations constructed for this problem takes the following form:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -2 & 1 & 0 & 0 & 0 & 0 \\ 0 & \dots & \dots & \dots & 0 & 0 & 0 \\ 0 & 0 & 1 & -2 & 1 & 0 & 0 \\ 0 & 0 & 0 & \dots & \dots & \dots & 0 \\ 0 & 0 & 0 & 0 & 1 & -2 & 1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} u_0 \\ u_1 \\ \dots \\ u_i \\ \dots \\ u_{N-1} \\ u_N \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \dots \\ 0 \\ \dots \\ 0 \\ h \end{bmatrix} \tag{4}$$

Figure 1 illustrates the general form of the matrix, with white fields marking zeros and colored fields standing for non-zero entries. Isogeometric collocation method (ISO-C) with quadratic B-splines and linear FEM both yield matrices of the same structure. The distribution of zero and non-zero entries is the same in all those cases (although the exact non-zero values may obviously differ). Various colors correspond to various frontal sub-matrices of the global matrix, which is indicated in the figure of the tree. For brevity, all the following tree figures end up on the root vertex and therefore the backward substitution trees developing rightwards from the root are skipped.

The structure of the matrix for one-dimensional (or two-dimensional, given a specific node numbering) finite difference method is identical to the structure of the matrix for finite element method with linear basis functions.

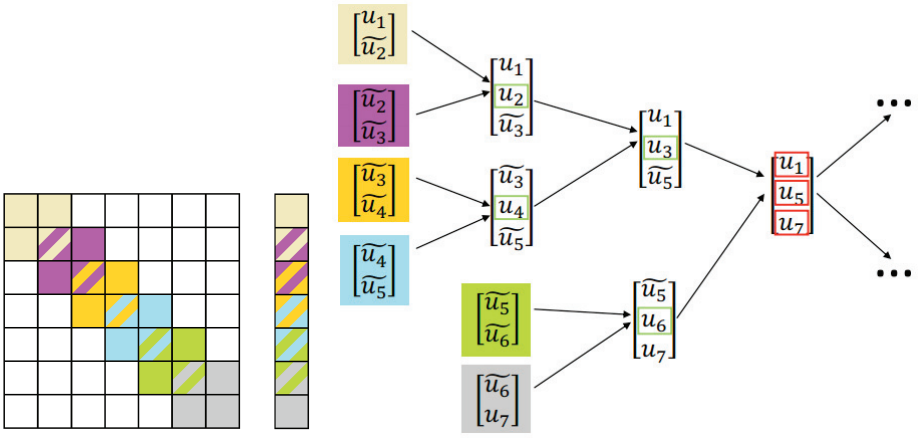

Figure 1: 1D FDM, linear FEM and ISO-C with quadratic B-splines. **Left panel:** Global matrix with frontal sub-matrices for $N = 7$. **Right panel:** Forward elimination and backward substitution tree in the multifrontal solver algorithm for the given matrix.

3 Finite element method with hierarchical basis functions

To apply one-dimensional FEM with bubble basis functions [3], the elliptic PDE (1) must be restated into an equivalent weak formulation:

$$\text{Find } u \in V = \{u \in H^1(0,1) : u(0) = 0\} \text{ such that} \tag{5}$$

$$b(v,u) = l(v), \forall v \in V \tag{6}$$

$$\text{where } b(u,v) = \int_0^1 \frac{dv(x)}{dx}\frac{du(x)}{dx}dx \tag{7}$$

$$l(v) = v(1) \tag{8}$$

Let us use hierarchical functions of an arbitrary order p. The global system of equations takes the following form:

$$\begin{bmatrix} 1 & 0 & 0 & \dots & & \\ b(e_1,e_0) & b(e_1,e_1) & b(e_1,e_2) & 0 & \dots & \\ b(e_2,e_0) & b(e_2,e_1) & b(e_2,e_2) & b(e_2,e_3) & b(e_2,e_4) & \dots \\ 0 & 0 & b(e_3,e_3) & b(e_3,e_3) & b(e_3,e_3) & \dots \\ 0 & 0 & b(e_4,e_3) & b(e_4,e_2) & b(e_4,e_4) & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \\ a_4 \\ \dots \end{bmatrix} = \begin{bmatrix} l(e_0) \\ l(e_1) \\ l(e_2) \\ l(e_3) \\ l(e_4) \\ \dots \end{bmatrix} \tag{9}$$

Figure 2 depicts contributions of the particular elements to the values in the matrix. The right panel shows the flow of the multifrontal solver algorithm in the forward elimination phase, including the initial phase of static condensation. The static condensation phase reduces the system to the one depicted in figure 1.

Field coloring indicates what particular finite elements contribute to the matrix entries.

Figure 4 illustrates the matrix for 2D FEM with quadratic basis functions. Notice that for the two-dimensional case the static condensation does not reduce the system to an instance of 2D FEM with linear basis functions (as in figure 3).

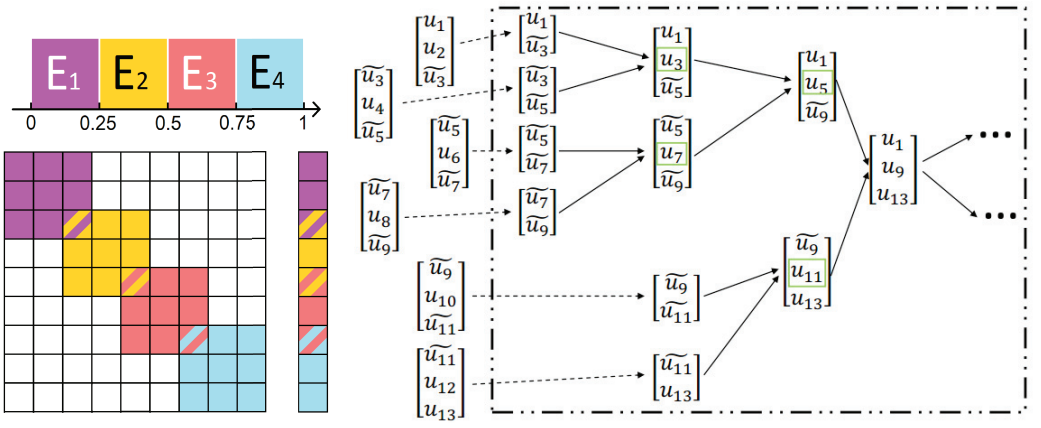

Figure 2: 1D finite element method with second order hierarchical basis functions. **Top left panel:** Mapping between the colors and the contributing elements. **Bottom left panel:** Global matrix with colors indicating contributions of particular elements for $N = 9$. **Right panel:** Flow of the multifrontal solver algorithm for $N = 13$. The dotted arrows indicate static condensation. Note that the highlighted subtree is equivalent to the forward elimination tree for FDM with $N = 7$, as in figure 1.

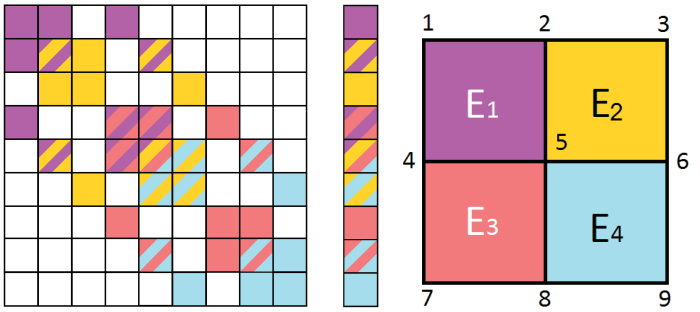

Figure 3: 2D finite element method with linear basis functions. **Left panel:** Global matrix with colors indicating contributions of particular elements. **Right panel:** Mapping between the colors and the contributing elements.

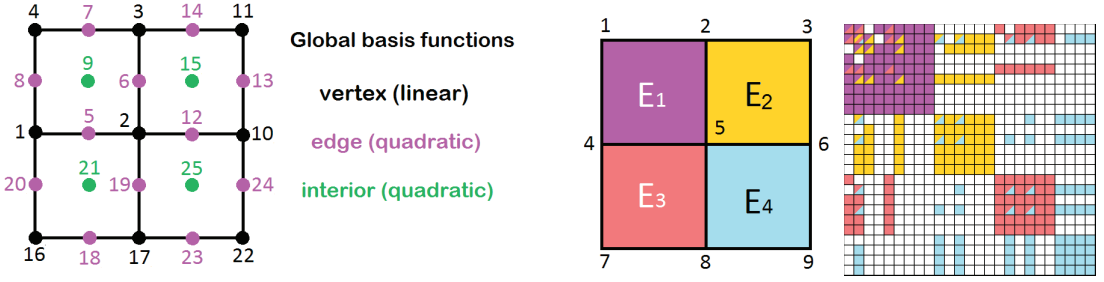

Figure 4: 2D finite element method with linear and quadratic hierarchical basis functions, including interior and edge basis functions. **Left panel:** Indexing of the vertex, edge and interior basis functions. **Middle panel:** Mapping between the colors and the contributing elements. **Right panel:** Global matrix with colors indicating contributions of particular elements.

4 Isogeometric collocation method with higher order B-splines

In general, for the isogeometric collocation method with B-splines of the order p, there are $N+p$ basis functions, and at each collocation point there are $p+1$ non-zero basis functions. This leads to the following equation system:

$$\sum_{i=1,\ldots,p+1} a_i \frac{d}{dx}\left(\frac{dN_{i,p}(c_1)}{dx}\right) = 0 \tag{10}$$

$$\ldots$$

$$\sum_{i=k,\ldots,k+p} a_i \frac{d}{dx}\left(\frac{dN_{i,p}(c_k)}{dx}\right) = 0 \tag{11}$$

$$\ldots$$

$$\sum_{i=N,\ldots,N+p} a_i \frac{d}{dx}\left(\frac{dN_{i,p}(c_N)}{dx}\right) = 0 \tag{12}$$

Note that there are $N+p$ unknowns and therefore p equations need to be added to the system above. Two of them can be obtained from the boundary conditions:

$$\sum_{i=1,\ldots,p+1} a_i N_{i,p} = 0 \tag{13}$$

$$\sum_{i=N,\ldots,N+p} a_i \frac{dN_{i,p}}{dx} = 1 \tag{14}$$

but to get the remaining $p-2$ equations, $p-2$ additional collocation points are still required. Assuming they are all located at the first element, the resulting multi-diagonal system of linear equations is as follows:

$$\begin{bmatrix} N_{1,p}(0) & \ldots & N_{N+p,p} & 0 & 0 & 0 & 0 \\ N''_{1,p}(c_1) & \ldots & N''_{p+1,p}(c_1) & 0 & 0 & 0 & 0 \\ \ldots & & & & & & \\ N''_{1,p}(c_p) & \ldots & N''_{p+1,p}(c_p) & 0 & 0 & 0 & 0 \\ 0 & \ldots & \ldots & \ldots & 0 & 0 & 0 \\ 0 & \ldots & N''_{k,p}(c_k) & \ldots & N''_{k+p,p}(c_i) & \ldots & 0 \\ 0 & 0 & 0 & \ldots & \ldots & \ldots & 0 \\ 0 & 0 & 0 & 0 & N''_{N,2}(c_{N+p-2}) & \ldots & N''_{N+p,p}(c_{N+p-2}) \\ 0 & 0 & 0 & 0 & 0 & \ldots & N'_{N+p,p}(1) \end{bmatrix} * \begin{bmatrix} a_1 \\ a_2 \\ \ldots \\ a_i \\ \ldots \\ a_{N+p-1} \\ a_{N+p} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \ldots \\ 0 \\ \ldots \\ 0 \\ 1 \end{bmatrix} \tag{15}$$

Figure 1, originally appearing in the context of FDM, shows as well the matrix generated for 1D quadratic ISO-C. Figure 5 shows the significantly more complex matrix for 2D quadratic ISO-C.

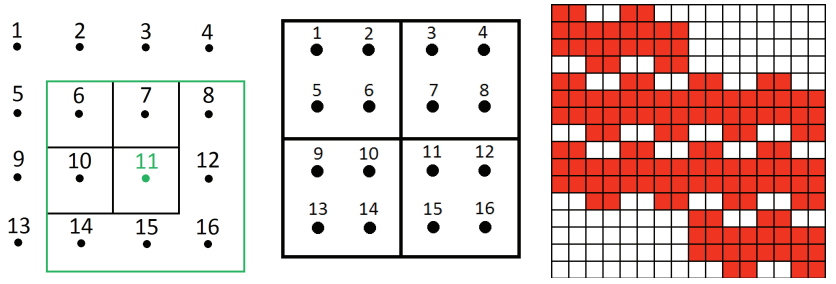

Figure 5: 2D isogeometric collocation method with quadratic B-spline basis functions. **Left panel:** Numbering of the B-spline functions (points indicating the geometric centers of their supports), with the support of sample B-spline (#11) marked green. **Middle panel:** Location and numbering of the collocation points. **Right panel:** Corresponding global matrix with red color indicating non-zero elements.

5 One-dimensional isogeometric finite element method

In the isogeometric FEM (ISO-FEM), the solution of (1-3) is approximated with B-spline basis functions:

$$u(x) \approx \sum_i N_{i,p}(x)\, a_i \qquad v \in \{N_{j,p}\}_j \tag{16}$$

where

$$N_{i,0}(\xi) = I_{[\xi_i,\xi_{i+1}]} \tag{17}$$

$$N_{i,p}(\xi) = \frac{\xi - \xi_i}{x_{i+p} - \xi_i} N_{i,p-1}(\xi) + \frac{\xi_{i+p+1} - \xi}{x_{i+p+1} - \xi_{i+1}} N_{i+1,p-1}(\xi) \tag{18}$$

where $I_{[\xi_i,\xi_{i+1}]}$ is the identity function over the interval $[\xi_i, \xi_{i+1}]$.

Substituting these definitions into the weak form allows to obtain the discrete weak formulation, which is stated as

$$\sum_i b(N_{j,p}(x), N_{i,p}(x))\, a_i = l(N_{j,p}(x)), \forall j \tag{19}$$

The finite elements are identified with the intervals:

$$\{E_i = [\xi_{i-1}, \xi_i] = [\frac{i-1}{N}, \frac{i}{N}]\}_{i=1,\ldots,N} \tag{20}$$

Let us focus on the case of quadratic B-splines:

$$u(x) \approx \sum_{i=1,\ldots,N+2} N_{i,2}(x)\, a_i \qquad v \in \{N_{j,2}\}_{j=1,\ldots,N+2} \tag{21}$$

This leads to the following system of linear equations:

$$\sum_{i=1,\ldots,N+2} b(N_{j,2}(x), N_{i,2}(x))\, a_i = l(N_{j,2}(x)), \forall j = 1,\ldots,N+2 \tag{22}$$

In the case of second order B-splines, each basis function has support over three finite elements. The main difference between one-dimensional FEM with second order bubble functions

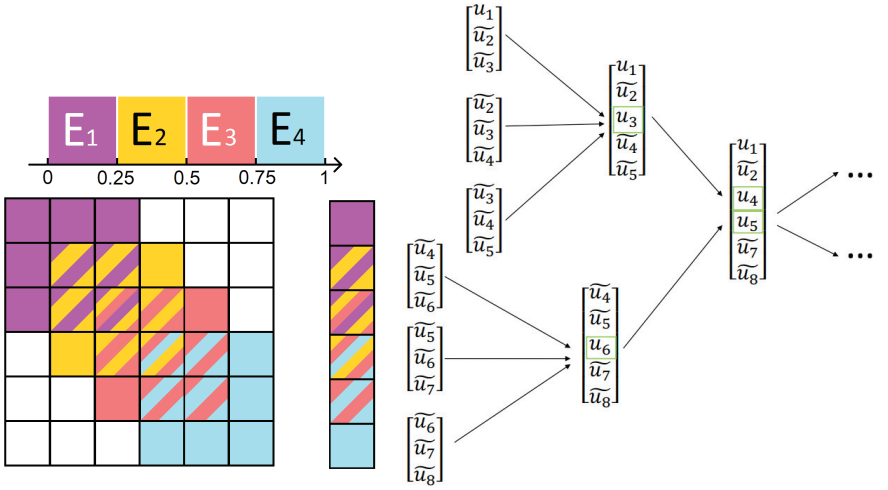

Figure 6: 1D isogeometric finite element method with quadratic B-splines and isogeometric collocation method with cubic B-splines. **Top left panel:** Mapping between the colors and the contributing elements. **Bottom left panel:** Global matrix for $N = 6$ with colors indicating contributions of particular elements (for ISO-FEM). **Right panel:** Forward elimination tree in the multifrontal solver algorithm for $N = 8$.

and the ISO-FEM with second order B-splines is that in the case of bubble functions it is possible to compress the frontal matrices by performing the static condensation [4], while in the case of B-spline basis functions this reduction is impossible.

The figure 6 depicts the resulting equation system and a tree for the multifrontal solver.

6 Numerical results

In this section we provide the experimental measurements for the integration as well as forward elimination and backward substitution phases of our multi-frontal direct solver on GPU, executed with linear, quadratic and cubic B-spline basis functions. The numerical results have been obtained on GeForce GTX 780 graphic card equipped with 3 gigabytes of memory and 2304 cores, as well as on NVIDIA Tesla K20c device, which has 5 gigabytes of memory and 2496 cores.

7 Conclusions

We can draw the following conclusions from the theoretical analysis of the structure of matrices for different methods as well as from the numerical experiments:

- Since the sparsity pattern of the matrix is identical for finite difference method (FDM), finite element method with linear basis functions, both hierarchical (FEM) and isogeometric (IGA-FEM), as well as with isogeometric collocation (ISO-C) method with quadratic B-splines, the same multifrontal solver implementation can be employed for all these cases.
- Since the sparsity pattern of the matrix is identical, the same multifrontal solver implementation can be employed for FDM, FEM with linear basis functions (both hierarchical

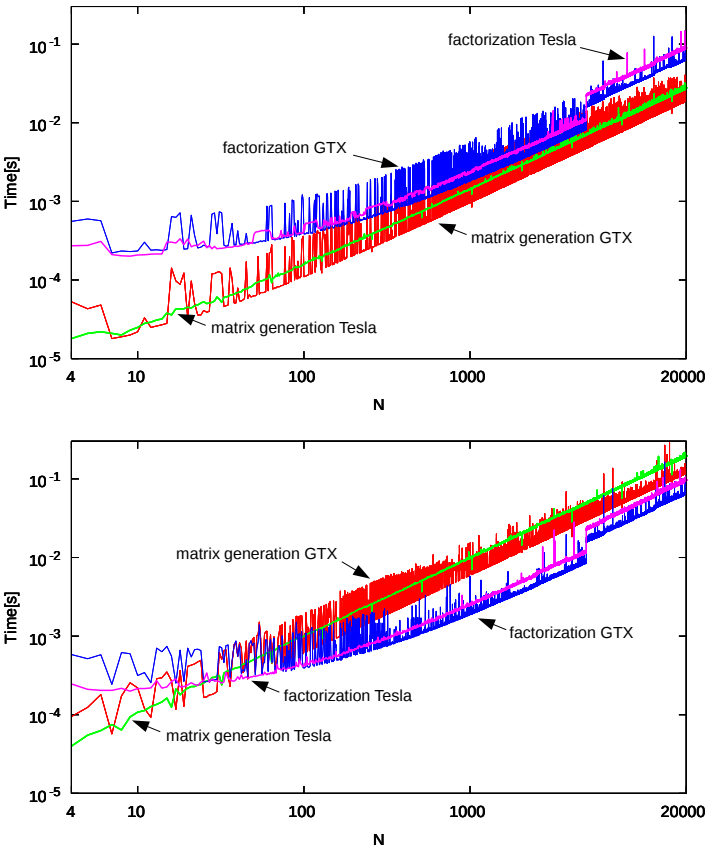

Figure 7: Comparison of integration and execution times of our GPU solver for one dimensional multi-frontal solver with isogeometric basis functions.
Top panel: 1D IGA FEM with linear basis functions integration, 1D IGA FEM with linear B-splines factorization. **Bottom panel:** 1D IGA FEM with quadratic basis functions integration, 1D IGA FEM with quadratic B-splines factorization.

FEM and isogeometric FEM), as well as ISO-C method with quadratic B-splines.

What particular method has been applied remains transparent for the solver as long as the method produces a matrix of the same sparsity pattern. This remains true in both 1D and 2D.

- Similarly, C^{p-1} IGA-FEM with p-order B-splines and IGA-C with $(p+1)$-order B-splines generate the same sparsity pattern of the global matrix.

- The IGA-C method, however, differs from the IGA-FEM method by not requiring the numerical integration step. As can be concluded from the numerical experiments, the integration constitutes a significant part of the execution time.

- The equation system resulting from FEM based on hierarchical functions in 1D can always be reduced to a tridiagonal matrix via the static condesation step. Thus, the cost of

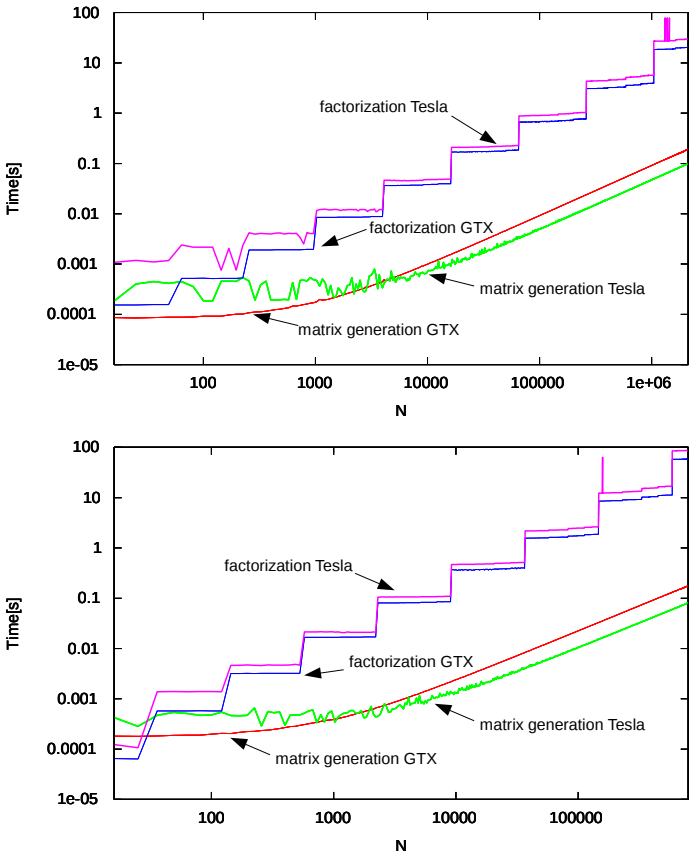

Figure 8: Comparison of integration and execution times of our GPU two dimensional multi-frontal solver with isogeometric basis functions. **Top panel:** 2D IGA FEM with linear basis functions integration, 2D IGA FEM with linear B-splines factorization. **Bottom panel:** 2D IGA FEM with quadratic basis functions integration, 2D IGA FEM with quadratic B-splines factorization. **Results:** In both cases factorization time jumps are strictly connected to excessing available number of cores/SMX-es on GPU. As B-spline order increases, matrices become larger and more dense, which leads to the increased matrix generation time.

the solver is equal to the 1D FEM solver with linear basis functions, plus the static condensation time.

- The static condensation in the 2D FEM with hierarchical basis functions does not reduce the system to equivalent to the 2D FEM with linear functions. Thus, the cost of the solver may be significantly different.

- The static condensation in the 1D and 2D IGA-FEM is not possible, since there are no local basis functions fully contained within a single element.

Acknowledgments The work presented in this paper was supported by Polish National Science Center grants no. DEC-2011/01/B/ST6/00674 and DEC-2011/03/B/ST6/01393.

References

[1] J.A. Cottrel, T. J. R. Hughes, J. Bazilevs, Isogeometric Analysis. Toward Integration of CAD and FEA, 2009

[2] F. Auricchio, L. Beirão da Veiga, T. J. R. Hughes, A. Reali, G. Sangalli, *Isogeometric collocation methods*, Mathematical Models and Methods in Applied Sciences, 20/11 (2010), 2075-2107

[3] L. Demkowicz, Computing with hp-adaptive finite element method, Part I. One and two dimensional elliptic and Maxwell problems with application, CRC Press, Taylor & Francis, 2007

[4] Paolo Bientinesi, V. Eijkhout, K. Kim, J. Kurtz, R. van de Geijn, Sparse Direct Factorizations through Unassembled Hyper-Matrices, Computer Methods in Applied Mechanics and Engineering, 01/2010

[5] K. Kuźnik, M. Paszyński, V. Calo, Grammar-Based Multi-Frontal Solver for One Dimensional Isogeometric Analysis with Multiple Right-Hand-Sides, Procedia Computer Science, 18 (2013) 1574-1583

[6] K. Kuźnik, M. Paszyński, V. Calo, Graph Grammar-Based Multi-Frontal Parallel Direct Solver for Two-Dimensional Isogeometric Analysis, Procedia Computer Science, 9 (2012) 1454-1463

[7] P. Lipski, M. M. Paszyński, *Multi-Frontal Parallel Direct Solver for One Dimensional Isogeometric Collocation Method*, Proceedings of the International Conference on Computer Methods in Computer Science, (2015)

[8] M. Woźniak, K. Kuźnik, M. Paszyński, V. M. Calo, D. Pardo, *Computational cost estimates for parallel shared memory isogeometric multi-frontal solvers*, Computers and Mathematics with Applications, 67(10) (2014) 1864-1883

[9] K. Kuźnik, M. Paszyński, V. Calo, *Graph grammar based multi-frontal solver for isogeometric analysis in 1D*, Computer Science, 14(4) (2013), 589-613

[10] M. Woźniak, M. Paszyński, D. Pardo, L. Dalcin, V. M. Calo, *Computational cost of isogeometric multi-frontal solvers on parallel distributed memory machines*, Computer Methods in Applied Mechanics and Engineering, 284, 2015, 971-987

[11] N. Collier, D. Pardo, L. Dalcin, M. Paszyński, V. M. Calo, *The cost of continuity: A study of the performance of isogeometric finite elements using direct solvers*, Computer Methods in Applied Mechanics and Engineering, 213216, 2012, 353-361

[12] D. Goik, K. Jopek, M. Paszyński, A. Lenharth, D. Nguyen, K. Pingali, *Graph Grammar based Multi-thread Multi-frontal Direct Solver with Galois Scheduler*, Procedia Computer Science, 29 (2014), 960-969