PROCEEDINGS of the
23rd International Congress on Acoustics
9 to 13 September 2019 in Aachen, Germany

Flexible multi-level fast multipole BEM with direct solver for industrial acoustic problems

Yue LI[1,2*], Onur ATAK[1], Wim DESMET[2,3]

[1]Siemens Industry Software, Interleuvenlaan 68, B-3001 Leuven, Belgium
[2]Noise and Vibration Research Group, PMA, KU Leuven, Celestijnenlaan 300 B, B-3001, Heverlee, Belgium
[3]DMMS core lab, Flanders Make, Belgium

Abstract

Fast Multipole Boundary Element Method (FMBEM) for acoustics is a very powerful method to solve ultra large problems, which exploits iterative solvers. However, for Helmholtz problem iterative solvers may fail to converge in some cases. Whereas FMBEM provides a unique offering for ultra large problems, many industrial problems in fact can be solved with tens of thousands of Dofs and can be considered as medium to large size. For such problems, direct solvers are still viable and provide a good option in terms of stability and convenience for solving many right-hand sides, especially in light of recent many core architectures like GPU. In this paper, we present a flexible fast multipole accelerated indirect BEM for solving medium to large size acoustic problems. The assembly process of the indirect BEM is accelerated by a flexible multi-level fast multipole method. The system matrix is then solved by a direct solver. Numerical results show that this combination offers an efficient and robust approach, where medium to large size acoustic problems can be efficiently solved on a modern desktop computer.

Keywords: Flexible FMBEM, Acoustics, Direct solver

1 INTRODUCTION

Boundary Element Method (BEM), sometimes referred to as boundary integral equation method (BIEM) or boundary integral solutions, is a well-known numerical method in many engineering applications. The unique advantage of BEM over other volume based numerical methods, e.g. Finite Element Method (FEM), Finite Volume Method (FVM), is that BEM reduces the dimensionality of the problem by one. It only requires a two-dimensional discretization for a three-dimensional problem. This advantage brings great benefit to Computational Aided Engineering (CAE) applications considering that the volume meshing process can be a significant cost in practice. Besides, unbounded domains are common for exterior acoustic applications. BEM is naturally convenient in computing unbounded problems as the Kirchhoff-Helmholtz integral equation satisfies the Sommerfield radiation condition automatically. The well-known limitation of conventional BEM (CBEM) lies in its fully populated coefficient matrix. The numerical complexity of CBEM is proportional to $O(N^2)$ in assembly and $O(N^3)$ in solving with conventional direct solvers, where N denotes the number of degree of freedoms (Dofs) in the model. Over the years, fast techniques such as fast multipole method [1], $\mathscr{H}$-matrix [2], pre-corrected FFT, have been developed to accelerate the conventional BEM. These techniques most often exploit iterative solvers with the consideration of saving memory cost for extremely large cases. Most of the existing FMBEM literature contribute to the direct BEM (dBEM) formulation [3], which limits the applications to closed boundary conditions. It should be noted that dBEM can only solve the Helmholtz equation in either interior or exterior domain at a time. Whereas various industrial cases require the modelling of open boundaries and/or combined interior-exterior problems. Indirect Galerkin BEM (iBEM) formulation is able to handle such cases. The attractive feature of employing a Galerkin approach is that the system of equations of the boundary element formulation becomes symmetric which reduces the computational cost by half.

*li.yue@siemens.com

Moreover, the existing FMBEM solution coupled with iterative solvers provides a unique offering for ultra large problems. Many industrial problems in fact can be solved with tens of thousands of Dofs and can be considered as medium to large size. For such problems, direct solvers are still viable and provide a good option in terms of stability and convenience of solving many right-hand sides. In addition, recent advances in many core architectures, such as GPUs, make dense solvers competitive again [7].

In this paper, we investigate the possibility of applying direct solver to FMBEM. We present a fast multipole accelerated indirect BEM for solving medium to large size industrial problems. Instead of using iterative solver in FMBEM, we assembly the full system matrix during the computation in order to benefit from the robust direct solvers. The contributions of this paper lie in two aspects:

1) For the first time, a flexible multi-level structure is applied to fast multipole BEM using indirect formulations;

2) We apply flexible fast multipole method to accelerate BEM matrix assembly. The system matrix is eventually solved by a robust direct solver. Numerical results have shown the advantages of our approach for the target problems.

2 THEORY

2.1 Indirect boundary integral formulation for Helmholtz problems

For time harmonic acoustic problems, the steady acoustic pressure p at any location $\boldsymbol{x}$ in a three dimensional fluid domain V is governed by Helmholtz differential equation

$$\nabla^2 p(\boldsymbol{x}) + k^2 p(\boldsymbol{x}) = 0, \quad \forall \boldsymbol{x} \in V \tag{1}$$

where $k = \omega/c$ is the acoustic wave number depending on angular frequency ω and speed of sound$c$.

In indirect BEM formula, the boundary variables are defined as double layer potential μ and single layer potential σ, which are defined as the difference of the pressure and the difference of the normal derivative of the pressure respectively:

$$\mu(\boldsymbol{y}) = p_1(\boldsymbol{y}^+) - p_2(\boldsymbol{y}^-) \tag{2}$$

$$\sigma(\boldsymbol{y}) = \frac{\partial p_1(\boldsymbol{y}^+)}{\partial n_{y^+}} - \frac{\partial p_2(\boldsymbol{y}^-)}{\partial n_{y^-}} = -j\rho\omega[q_1(\boldsymbol{y}^+) + q_2(\boldsymbol{y}^-)] \tag{3}$$

where the subscripts 1 and $+$ are associated with the direction of the unit normal, 2 and $-$ are associated with the opposite direction of the unit normal.

Using the single layer and double layer potentials, we can write the indirect Kirchoff-Helmholtz integral equation as follows:

$$p(\boldsymbol{x}) = \int_S [\frac{\partial G(\boldsymbol{x},\boldsymbol{y})}{\partial n_y}\mu(\boldsymbol{y}) - G(\boldsymbol{x},\boldsymbol{y})\sigma(\boldsymbol{y})]\,\mathrm{d}S_y, \quad \forall \boldsymbol{x},\boldsymbol{y} \in S \tag{4}$$

where n_y denotes unit normal at a source point $\boldsymbol{y}$. $G(\boldsymbol{x},\boldsymbol{y})$ is the Green function of the Helmholtz equation. Its 3D form is given as

$$G(\boldsymbol{x},\boldsymbol{y}) = \frac{e^{ik|\boldsymbol{x}-\boldsymbol{y}|}}{4\pi\,|\,\boldsymbol{x}-\boldsymbol{y}\,|} \tag{5}$$

The associated boundary conditions of Helmholtz equation can be written for both sides as well. Assuming a thin boundary surface, it leads to the following boundary condition expressions:

$$\mu(\boldsymbol{y}) = 0, \quad p(\boldsymbol{y}) = \overline{p}(\boldsymbol{y}) \;\; on \;\; S_p \tag{6}$$

$$\sigma(\boldsymbol{y}) = 0, \quad \frac{\partial p(\boldsymbol{y})}{\partial n} = -j\rho_0\omega\overline{v}_n(\boldsymbol{y}) \;\; on \;\; S_v \tag{7}$$

$$\sigma(\boldsymbol{y}) = -jk\beta\mu(\boldsymbol{y}), \quad \frac{\partial p(\boldsymbol{y})}{\partial n} = -jk\beta p(\boldsymbol{y}) \ \ on \ \ S_Z \tag{8}$$

At each position on the boundary sureface, either the double layer potential is zero or the single layer potential is zero, or the single layer potential is related to the double layer potential. From these relations the indirect boundary integral formulation can be reformulated as:

$$p(\boldsymbol{x}) = -\int_{S_p} \sigma(\boldsymbol{y}) G(\boldsymbol{x},\boldsymbol{y}) \mathrm{d}S_p + \int_{S_v} \mu(\boldsymbol{y}) \frac{\partial G(\boldsymbol{x},\boldsymbol{y})}{\partial n} \mathrm{d}S_v + \int_{S_Z} \mu(\boldsymbol{y}) [\frac{\partial G(\boldsymbol{x},\boldsymbol{y})}{\partial n} + jk\beta(\boldsymbol{y}) G(\boldsymbol{x},\boldsymbol{y})] \mathrm{d}S_Z \tag{9}$$

Subsequently, Eq.(9) can be used to rewrite the boundary condition in equations Eq.(6)-Eq.(8). They can be written in a compact form by using a function f of unknowns σ and μ.

$$f_p(\sigma,\mu) = p \ \ on \ \ S_p \tag{10}$$

$$f_v(\sigma,\mu) = -j\rho_0\omega v_n \ \ on \ \ S_v \tag{11}$$

$$f_Z(\sigma,\mu) = 0 \ \ on \ \ S_Z \tag{12}$$

A variational approach is adopted to avoid Hadamard finite part integrals and to achieve symmetric system matrices. The equivalent variational statement of the equations Eq.(10)-Eq.(12) can be written as [6]:

$$\forall(\delta\sigma,\delta\mu): \int_{S_p} f_p(\sigma,\mu)\delta\sigma \mathrm{d}S_p + \int_{S_v} f_v(\sigma,\mu)\delta\mu \mathrm{d}S_v + \int_{S_Z} f_Z(\sigma,\mu)\delta\mu \mathrm{d}S_Z = \int_{S_p} \overline{p}\delta\sigma \mathrm{d}S_p - \int_{S_v} j\rho_0\omega\overline{v}_n\delta\mu \mathrm{d}S_v \tag{13}$$

2.2 Fast multipole method

The diagonal fast multipole form is proposed by [1]. The basic idea of fast multipole method is to convert the point-to-point operation to a cluster-to-cluster operation. Considering two clusters of points as shown in Fig.1, x_0 and y_0 are the centers of the source cluster C_1 and the target cluster C_2 respectively. When C_1 and C_2 are well separated, one could convert a direct evaluation from x_j to y_l, to a multipole evaluation which consists of three main steps: 1) Upwardpass: the contribution of x_j is firstly computed to x_0, the center of the source cluster C_1; 2) Multipole translation: compute the multipole translation from source cluster's center to target cluster's center, from x_0 to y_0 in this case; 3) Downwardpass: distribute the contribution from center y_0 to y_l in the cluster C_2.

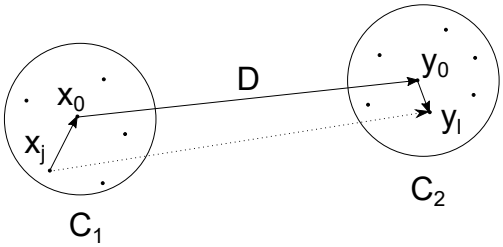

Figure 1. Clusters of x_j (C_1) and y_l (C_2)

In mathematical perspective, the basic solution of the Helmholtz equation (Eq.(5)) can be rewritten using Gegenbauer addition theorem and plane wave expansion form:

$$\frac{e^{ik\|x_j - y_l\|}}{4\pi \parallel x_j - y_l \parallel} = \frac{e^{ik\|\boldsymbol{d}+\boldsymbol{D}\|}}{4\pi \parallel \boldsymbol{d}+\boldsymbol{D} \parallel} = \frac{ik}{16\pi^2} \int_{\mathbb{S}^2} \mathrm{e}^{ik(y_l - y_0)\cdot\boldsymbol{s}} M_L(\boldsymbol{s},\boldsymbol{D}) e^{-ik(x_j - x_0)\cdot\boldsymbol{s}} \mathrm{d}s \tag{14}$$

The diagonal translation operator M_L is defined as

$$M_L(\boldsymbol{s},\boldsymbol{D}) = \sum_{l=0}^{L-1}(2l+1)i^l h_l^{(1)}(kD)P_l(\boldsymbol{s}\cdot\hat{\boldsymbol{D}}) \tag{15}$$

where $\boldsymbol{d} = (y_l - y_0) - (x_j - x_0)$, $\boldsymbol{D} = y_0 - x_0$, $d = \| \boldsymbol{d} \|$, $D = \| \boldsymbol{D} \|$, $h_l^{(1)}(\cdot)$ is a spherical Hankel function of the first kind of order 1, $P_l(\cdot)$ is a Legendre polynomial. Normalized vectors are indicated by $\hat{(\cdot)} = (\cdot)/(\| \cdot \|)$. $\boldsymbol{s}$ is the unit sphere given by $\boldsymbol{s}(\theta,\phi) = (sin\theta cos\phi, sin\theta sin\phi, cos\theta)$. $0 \le \theta \le \pi$, $0 \le \phi \le 2\pi$ are polar angle and azimuthal angle in the spherical coordinate system.

The integral over the unit sphere surface $\mathbb{S}$ can be evaluated numerically: in the θ direction by using a Gauss-legendre integration with L points and in the ϕ direction by using a trapezoidal rule with $2L$-points.

$$\int_{\mathbb{S}^2} f(s)\mathrm{d}s = \int_{-1}^{1}\int_{0}^{2\pi} f(s)\mathrm{d}\phi\mathrm{d}(cos(\theta)) \approx \sum_{i=1}^{L}\sum_{j=1}^{2L}\omega_i\frac{\pi}{L}f(s_{i,j}) \tag{16}$$

with ω_i the weight of the Gauss-Legendre integration for the ith integration point in the θ-direction.

In numerical implementation, the sum of l is truncated at $l = L$. A semi-empirical rule for the expansion length is given by many papers

$$L = kd_l + p\log(kd_l + \pi) \tag{17}$$

where p specifies the required precision, d_l is the maximum diameter of the clusters.

2.3 Flexible multi-level FMiBEM

When applying FMM to BEM, a partition algorithm is required to cluster the elements of the model into groups. This can either be done on one level structure resulting in a single-level scheme, or on many levels resulting in a multi-level scheme. In single-level FMBEM, the number of the clusters should be approximately equal to the root of the number of elements to ensure the efficiency of the algorithm [5]. As the size of the problem increases, the number of the elements in the near field clusters increases proportionally to the root of the total number of elements, which makes the single-level FMBEM less efficient. Multi-level scheme tackles this issue by employing a hierarchical structure. The number of elements in the cluster on the leaf level (lowest level) becomes independent of the total number of elements. The size of the model is then reflected by the number of levels in the tree structure. However, the cluster size increases from lower to higher level. Thus the expansion order increases due to its dependence on cluster diameter. The computational cost on upper levels increases quadratically with the expansion order with the consideration of employing more spherical Gauss points in θ and ϕ directions in Eq.(16). This becomes significant when the model size is getting larger, where more levels (5-10) are required.

Many of the existing work tackle this issue by employing an interpolation/filtering algorithm among different levels. Various schemes have been proposed in the literature [4]. The complexity of such algorithm is crucial in Helmholtz (and Maxwell) problems to ensure that it never overtakes the complexity of fast multipole method itself. Nevertheless, this adds extra costs to the entire computation. A direct multi-level approach is presented in [3], where the far field contributions are computed and stored on each level explicitly in order to avoid the interpolation/filtering procedure. By sacrificing some memory cost, a Burton-Miller collocational BEM with direct multi-level approach achieves better efficiency over the one with the interpolation/filter algorithm [3]. Inspired by the direct multi-level concept, we further develop a flexible and efficient *a priori* Octree structure for symmetric Galerkin indirect BEM.

As used by many FMBEM papers, the original Octree structure always starts with one bounding box, which generally is an over-determined box. Marked as level 0, the parent box is further divided into 2^d child boxes forming the next level. The procedure is repeated recursively till the predefined criteria is fulfilled. The upper levels of the original tree are determined by the structured partition. In contrast, *a priori* Octree starts with a

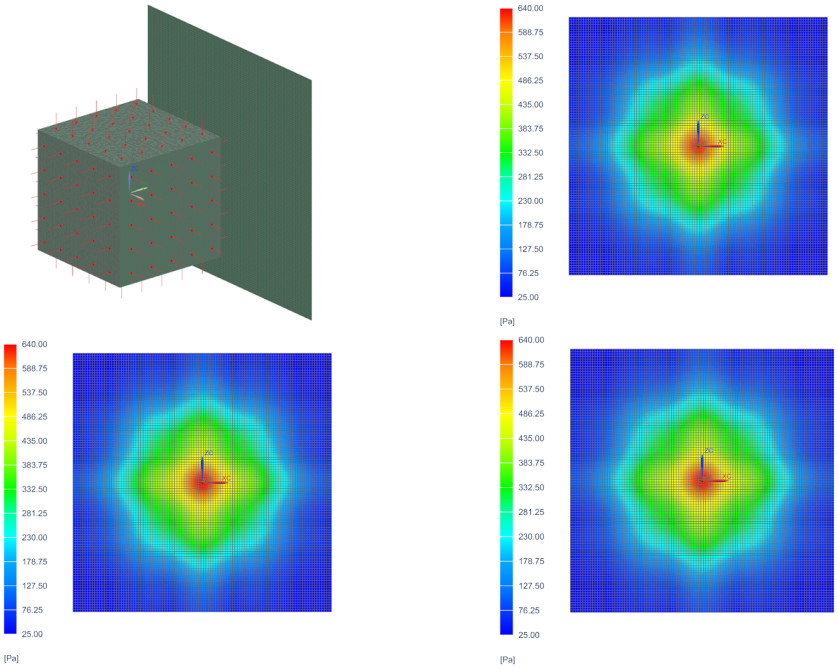

Figure 2. Cube mesh (17k Dofs) with field points (up left); Pressure plot of CBEM (up right); Pressure plot of FMBEM (down left); Pressure plot of f-FMaccBEM (down right).

level which has been computed *a priori*. Unlike the immutable structure in the original Octree, both the leaf level and the top level in *a priori* Octree are flexible and controllable, which provide an efficient multi-level structure for fast multipole computation. The expansion order on the top level is maintained as low as possible to ensure the efficiency of the ML calculation, and to avoid the interpolation/filtering algorithm at the same time. The leaf level is computed *a priori* to fulfill the accuracy requirement from multipole assumption. In the high frequency fast multipole formula, it is well-known that when kD becomes small (e.g. $kD < 0.09$ for expansion order 8), the spherical Hankel function can easily blow up to the magnitude of 10^{16}, which is the machine limit that double precision can achieve. *A priori* Octree allows a flexible construction of the tree structure. By estimating the box size on the leaf level *a priori*, it restrains the Hankel function within a reasonable range.

2.4 Hyper-singularity

Hyper-singularity appears in both indirect BEM and Burton-Miller direct BEM formulas. There exists several ways to handle hyper-singularity, which includes singularity cancellation techniques, specialized quadrature methods and regularization techniques. The presented paper focuses on the application of practical regularization techniques to FMiBEM. Stokes' theorem is applied to transform hyper-singularity to singular and weakly singular form. Since the Octree structure has split our problem into near fields and far fields, the hyper-singularity shall be considered separately. Apparently, there is no singularity exist in the far fields. Thus the far field contributions are computed directly by the original hyper-singular form, while the near field is transformed by applying regularization technique. However, the regularization technique is only valid for closed boundary condition, or when the principle term of hyper-singularity is zero. In practical implementation, this introduces additional calculations on the edges in the near field clusters. Special care must be paid to the clusters which contains shared edges.

2.5 Direct solver

Most of the existing FMBEM papers exploit various iterative solvers to solve ultra large problems. However, for Helmholtz problems, iterative solvers often have slow convergence, even may fail to converge in certain cases. Finding an efficient preconditioner remains challenging. Whereas FMBEM provides a unique offering for ultra large problems, many industrial problems are in fact can be solved with tens of thousands of Dofs. For such problems, direct solvers are still viable and provides a good option in terms of stability and convenience of solving many right-hand sides. In addition, dense solvers can now be efficiently accelerated by recent many core architectures such as GPUs [7]. Therefore, we investigate the possibility of combining FMBEM with direct solvers. We apply fast multipole method to accelerate the conventional BEM assembly procedure. Instead of using low rank approximation and iterative solvers, we compute the far field contributions for each cluster pairs and assembly back to the global system matrix. In this sense, the global system matrix can be solved by direct solvers. A dense Cholesky solver is used in our study. Benefiting from the symmetric feature of Galerkin approach, we save half of the far field computations, as well as the near field computations. Based on our tests, modern laptops/workstations are able to provide sufficient memory to handle the target size problems.

3 NUMERICAL EXAMPLES

All the following calculations are conducted using single thread CPU on a laptop with i7-7920HQ CPU and 32GB memory in Windows 10. In the following discussions, CBEM denotes conventioanl BEM, FMBEM denotes conventional fast multipole indirect BEM with iterative solver, f-FMaccBEM denotes flexible fast multipole accelerated indirect BEM with direct solver. We choose Cholesky solver for f-FMaccBEM, and Generalized Conjugate Residual (GCR) solver with Sparse Approximate Inverse (SPAI) preconditioner for FMBEM.

3.1 3D cube case

Firstly we consider a 3D cube case with prescribed velocity boundary condition. Excitation frequency is $800Hz$, giving $kA = 14.8$. The cube is discretized by linear triangular elements. Figure 2 shows the mesh of 17k Dofs together with field points. The contour plots of the field points' absolute pressure from CBEM, FMBEM and f-FMaccBEM are shown using the same scale in the same figure.

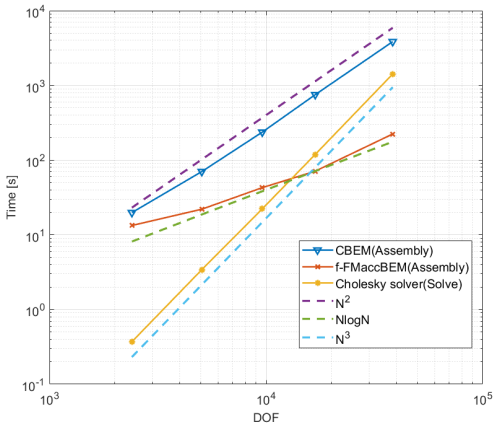

Figure 3. Assembly time of CBEM and f-FMaccBEM, and direct solver time

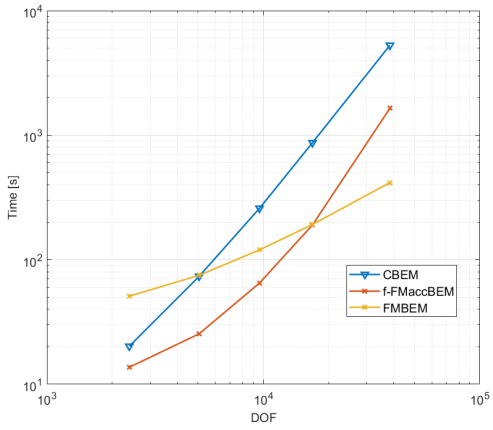

Figure 4. Total computational time for CBEM, FMBEM and f-FMaccBEM

Figure 3 plots the assembly time for CBEM and FMaccBEM together with the time of Cholesky direct solver. It can be found that the assembly complexity of f-FMaccBEM follows $NlogN$ compared to N^2 of the CBEM.

For the model size of 38k Dofs, we see a speed up factor of 17 from CBEM (1.1 *hours*) to f-FMaccBEM (3.7 *minutes*). It is of interest to observe that the cost of assembling far field matrix and inserting it back into the global system matrix doesn't influence the total complexity. For model size up to 20k Dofs, assembly cost is dominant in the total f-FMaccBEM computation. Significant speedup is achieved by f-FMaccBEM over CBEM. After 20k Dofs, the cost of Cholesky solver is taking over and becomes the dominant cost. For larger models, parallel computing techniques [7] can be applied to direct solver to obtain a good overall efficiency. Figure 4 shows the total computational time of CBEM, FMBEM and f-FMaccBEM. For small to medium size problem, we find the benefit of combining f-FMaccBEM with direct solver. It outperforms CBEM and the conventional FMBEM with iterative solver. Meanwhile, it should be noted that for a simple geometry in this test, the iterative solver can easily reach target tolerance within 15-20 iterations. For more complex geometries at high frequencies, iterative solvers may struggle to converge, which will be demonstrated in the next case.

3.2 Realistic car case

A Chrysler Neon car is studied below. The car mesh consists of 40,234 linear triangular elements and 20,010 nodes. As shown in figure 5, the engine of the car has been modelled in details. In this study we want to evaluate the frequency response function (FRF) of the engine noise. The engine is described by a unit velocity boundary condition, while the rest of the car mesh is sound hard boundary. Typically, the peak part of engine noise concentrates in the range $100-400Hz$. Figure 6 plots the FRF computed by CBEM and f-FMaccBEM. The receiver point is placed 50cm away from the driver's door. By taking CBEM as reference, the relative L_2 error of f-FMaccBEM is well controlled under 0.5% for the entire FRF.

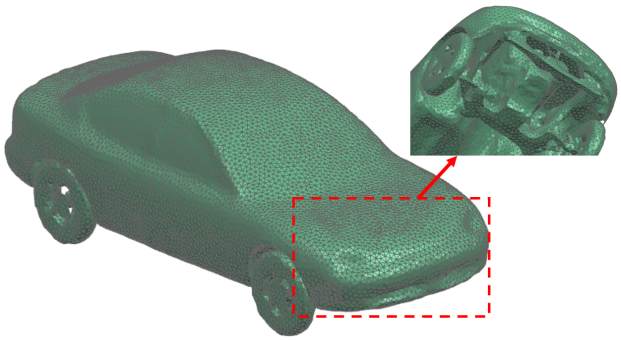

Figure 5. Car mesh

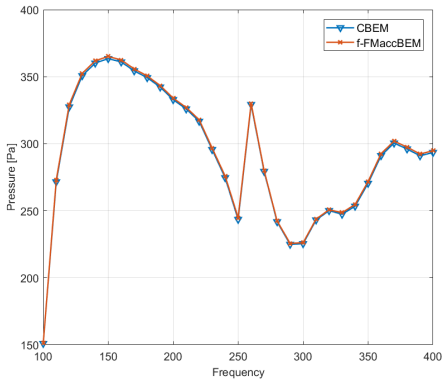

Figure 6. Frequency response function

Table 1. Computational time for the entire FRF

Solver	Assembly time	Solve time	Total time [mins]
CBEM	485	190	675
FMBEM	-	-	400
f-FMaccBEM	91	190	281

Table 1 gives the total computational time (exclude post processing) for the entire FRF. f-FMaccBEM saves 60% of time compared to CBEM, and 30% of time compared to FMBEM. Table 2 lists some detailed costs of FMBEM and f-FMaccBEM at different frequencies. f-FMaccBEM outperforms FMBEM at all frequencies. It can be seen that for higher frequencies, the iterative solver in FMBEM is getting difficult to converge. By using six element per wave length, the excitation frequency is valid up to $820Hz$. Although equipped a preconditioner,

the iterative solver requires 549 iterations to converge to the predefined tolerance 0.001 at $800Hz$. In contrast, the direct solver used by f-FMaccBEM remains efficient for all valid frequencies.

Table 2. Computational time of FMBEM and f-FMaccBEM

Solver	Frequency	Assembly time [s]	Solve time [s]	Total time[s]	Nr of iterations
FMBEM	100Hz	-	-	863	40
FMBEM	200Hz	-	-	707	75
FMBEM	400Hz	-	-	824	238
FMBEM	800Hz	-	-	1520	549
f-FMaccBEM	100Hz	219	358	577	-
f-FMaccBEM	200Hz	155	368	523	-
f-FMaccBEM	400Hz	164	365	529	-
f-FMaccBEM	800Hz	323	342	665	-

4 CONCLUSION

In this paper, we present a fast multipole accelerated indirect BEM for solving medium to large size acoustic problems. We investigate and successfully combine a flexible multi-level FMBEM with direct solver. The assembly process of the indirect BEM is accelerated by a flexible multi-level fast multipole method. The system matrix is then solved by a direct solver. Numerical results have shown that this combination offers a robust and fast solution, where medium to large size acoustic problems can be efficiently solved on a modern desktop computer. Future work will report on taking advantages of dense solver on modern hardware architectures such as GPUs.

ACKNOWLEDGEMENTS

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 721536.

REFERENCES

[1] Rokhlin, Vladimir. Diagonal forms of translation operators for the Helmholtz equation in three dimensions, Applied and Computational Harmonic Analysis, Vol 1 (1), 1993, pp 82-93.

[2] Hackbusch, Wolfgang. Hierarchical matrices: algorithms and analysis. Vol. 49. Springer, Heidelberg, 2015.

[3] Z.-S. Chen, H. Waubke, W. Kreuzer, A formulation of the fast multipole boundary element method (fmbem) for acoustic radiation and scattering from three-dimensional structures, Journal of Computational Acoustics, Vol 16, 2008, pp 303–320.

[4] E. Darve, The fast multipole method: numerical implementation, Journal of Computational Physics, Vol 160, 2000, pp 195–240.

[5] K. Giebermann, Fast summations methods of numerical solution of integral equations for scattering problems in R3, PhD thesis, 1997.

[6] T. Wu, Boundary Element Acoustics, Fundamentals and Computer Codes, WIT Press, 2000.

[7] Dongarra J, Gates M, Haidar A, Kurzak J, Luszczek P, Tomov S, Yamazaki I. Accelerating numerical dense linear algebra calculations with GPUs. In Numerical computations with GPUs, 2014, pp. 3-28.