RAL-TR-1999-039

Application of a domain decomposition method with Lagrange multipliers to acoustic problems arising from the automotive industry

Frédéric Magoulès [1] Karl Meerbergen [2] [3]
Jean-Pierre Coyette [2] [4]

ABSTRACT
The Finite Element Tearing and Interconnecting method for the Helmholtz equation is a recent non-overlapping domain decomposition method for solving linear systems arising from the finite element discretization of Helmholtz problems in bounded domains. This method was validated on two-dimensional external problems with first order absorbing boundary conditions. The purpose of this paper is to study the robustness and efficiency of iterative methods for the solution of the associated interface problem for three-dimensional interior problems arising from the automotive industry.

Keywords: Helmholtz equation, domain decomposition, iterative methods, parallel computing

AMS (MOS) subject classifications: 76Q05,65M55,65N55,65F50

[1] ONERA, 29 avenue de la Division Leclerc B.P. 72, 92322 Chatillon Cedex, France.

[2] LMS International, Interleuvenlaan 70, 3001 Leuven, Belgium.

[3] Current address : Rutherford Appleton Laboratory, Computational Science and Engineering Department, Chilton, Didcot OX11 0QX, United Kingdom.

[4] Current address : Free Field Technologies, 5/501 rue Charlemagne, 1238 Louvain-la-Neuve, Belgium.

Computational Science and Engineering Department
Atlas Centre
Rutherford Appleton Laboratory
Oxon OX11 0QX

June 8, 1999

Contents

1 Introduction

The motivation behind this work is the design of acoustically performant products, which is a major concern in many industrial sectors, and more specifically in automotive companies. This process relies on the use of various acoustic models. In the present study, uncoupled acoustic models for closed cavities are considered. The acoustic problem is 'decoupled' from the surrounding structure, i.e. the pressure field is assumed not to interact with the enclosing structure.

In its simplest linear form, the acoustic problem is governed, in the frequency domain, by the Helmholtz equation with suitable boundary conditions (Pierce 1981). The main unknown is the acoustic pressure field while boundary conditions are related to (Dirichlet) pressure constraints, (Neumann) normal pressure gradient constraints and (Robin) normal admittance constraints. The solution of such a problem is usually computed for a frequency range. This allows the acoustic response at particular locations within the cavity to be assessed. For a car compartment, these are usually the driver's and passenger's ears. The practical evaluation relies on the use of appropriate numerical models. For complex automotive geometries, boundary element and finite element methods offer the required flexibility. For finite element methods, some mesh requirements should be taken into account (around ten nodes per wavelength) which leads to huge mesh sizes when the dimension of the structure is proportional to some wavelength.

For industrial applications, the linear system obtained after a finite element discretization is usually solved with direct solvers for reasons of robustness. Unfortunately, the memory requirements and computational cost grow rapidly with the size of the acoustic model. In order to be able to solve larger models, parallel computers are employed. The parallel solution by direct methods is an option (Duff 1998). Iterative methods are often easier to parallelize and require less memory but may suffer from lack of robustness (Magoulès, Roux, Coyette and Lecomte 1998*b*). Another approach, called domain decomposition, relies on the decomposition of the entire domain into subdomains, so that the global problem is decomposed in a number of local problems, which can be solved independently. Because of this property, domain decomposition is well-suited for parallel computing. In the case of non-overlapping domain decomposition and in order to restore the connection between the subdomains, boundary conditions are imposed on the interfaces between the subdomains. This leads to a so-called interface problem that describes the coupling of the subdomains. The solution of this interface problem readily produces the solution on the global domain. A particular non-overlapping domain decomposition method called Finite Element Tearing and Interconnecting (FETI) (Farhat and Roux 1992) uses a direct solver for the local problems but an iterative one for the interface problem. The purpose of the method is to combine the robustness of a direct method with the flexibility of an iterative procedure.

The one-level FETI method for the Helmholtz equation with two Lagrange multipliers (Magoulès, Roux and de La Bourdonnaye 1998*a*) was initially developed for exterior Helmholtz problems, and has shown its performance for huge linear systems. The aim of this paper is to illustrate the robustness and efficiency of the method for three-dimensional internal Helmholtz problems with relatively small dimension. The theoretical computational cost is also derived. The method is much easier to implement in an existing code than the preconditioned method (Farhat, Macedo, Lesoinne, Roux, Magoulès and Bourdonnaye n.d.b). Moreover, its performance is expected to improve with the use of a global preconditioner based on Krylov spaces, as already shown for two-dimensional problems (de La Bourdonnaye, Farhat, Macedo, Magoulès and Roux 1998).

The plan of the paper is as follows. In 2, the mathematical formulation of the original problem is presented. In 3, the concept of the FETI method is introduced for the Helmholtz equation with two Lagrange multipliers and iterative solvers for the interface problem are also discussed. Section 4 deals with implementational aspects for the integration in an acoustic finite element code. In 5, convergence results for a number of test cases are shown. The first example is concerned with the design of a car compartment and the second is related to the prediction of transmission characteristics of an exhaust system. The main conclusions are formulated in 6.

2 Mathematical formulation

In this section, the problem is formulated and its variational formulation and discretization are presented. The general Helmholtz problem for the acoustic pressure u, in a bounded domain Ω with boundary conditions on $\partial\Omega$ can be written as follows: for $f \in L^2(\Omega)$ and $g \in L^2(\partial\Omega)$, find $u \in H^1(\Omega)$ such that

$$\begin{aligned} -\nabla^2 u - k^2 u &= f \qquad && \text{in } \Omega \\ \frac{\partial u}{\partial \nu} + \alpha u &= g \qquad && \text{on } \partial\Omega \end{aligned}$$

where k denotes the wave number, ν the unit outward normal on $\partial\Omega$, and α is a scalar. The boundary conditions are chosen in such a way that this problem is well-posed and has a unique solution. The boundary conditions are often defined as homogeneous or non-homogeneous Neumann boundary conditions (case $\alpha = 0$) which are related for example to a rigid body if $g = 0$, or Robin boundary conditions (case $\alpha \neq 0$) as induced by specific acoustic treatments such as absorbing materials. It is important to note that in the case of absorbing materials, the scalar α usually is complex, which implies complex arithmetic. Dirichlet boundary conditions are not considered here, because they are not often used in industrial simulations.

The variational formulation of this problem can be written as follows : for $f \in L^2(\Omega)$ and $g \in L^2(\partial\Omega)$, find $u \in H^1(\Omega)$ such that

$$\forall v \in H^1(\Omega) \ , \quad \int_\Omega (\nabla u \nabla v - k^2 uv) + \alpha \int_{\partial\Omega} uv = \int_\Omega fv + \int_{\partial\Omega} gv$$

The equivalence between the variational formulation and the initial hyperbolic problem can be found in (Lions and Dautray 1985). After Galerkin discretization with finite elements, the linear system

$$\tilde{K}x = b \tag{1}$$

is obtained where b denotes the right-hand side, and $\tilde{K}$ the sparse matrix defined by

$$\tilde{K} = K - k^2 M + \alpha M_R$$

where K is the stiffness matrix, M the mass matrix, and M_R the contribution from Robin boundary conditions along $\partial\Omega$. Since α is complex, $\tilde{K}$ is, in general, complex symmetric i.e. non-Hermitian.

3 A domain decomposition method for acoustics

In this section, the Finite Element Tearing and Interconnecting method for the Helmholtz equation is presented and the associated interface problem is derived. Iterative solvers for the interface problem are briefly discussed.

Various domain decomposition techniques exist and the reader is referred to (Roux 1995) for more details about sub-structuring methods. The idea behind the primal Schur complement method (Tallec 1994) and the dual Schur complement method (Farhat and Roux 1994) consists in solving independent problems on each subdomain with an additional constraint that forces continuity of the pressure u and pressure normal derivative $\partial u/\partial\nu$ along the interface. Both methods have been proven effective for coercive elliptic partial differential equations but serious difficulties are encountered for non-coercive elliptic problems, including the Helmholtz equation when the wave number k becomes an eigenvalue of the Laplacian operator. The original method (Després 1990) is based on a non-overlapping additive Schwarz algorithm and consists of the addition of a Robin boundary condition on the subdomains interfaces. This method has further been improved by the optimal choice of the coefficients (Chevalier and Nataf 1998) and extended to non-conforming meshes. A non-overlapping multiplicative Schwarz algorithm has been developed (Collino 1993).

The Finite Element Tearing and Interconnecting method for the Helmholtz equation is based on the Finite Element Tearing and Interconnecting (FETI) method for structural problems (Farhat and Roux 1992). The interface boundary conditions are modified so that after discretization, the local Helmholtz matrix in each subdomain does not become singular for some wave number k. This method can be derived with one Lagrange multiplier (Farhat, Macedo and Lesoinne n.d.*a*) (FETI-H) or with two Lagrange multipliers (Magoulès et al. 1998*a*) (FETI-H2LM). Note that this last formulation, coming from an augmented Lagrangian formulation, can be interpreted as a reformulation with two Lagrange multipliers of the original algorithm developed in (Després 1990). The key point is that for two-dimensional applications both formulations present the same dependency on frequency, mesh size, and number and shape of subdomains (Farhat et al. n.d.*b*).

3.1 Finite element tearing and interconnecting

The FETI-H2LM for the Helmholtz equation can be defined for two subdomains as follows. Let the domain Ω be decomposed into two non-overlapping subdomains Ω_s, $s = 1, 2$; let Γ_I denote the interface $\Gamma_I = \partial\Omega_1 \cap \partial\Omega_2$. Then for $f \in L^2(\Omega)$ and $g \in L^2(\partial\Omega)$, find $u_s \in H^1(\Omega_s)$ so that

$$\begin{aligned} -\nabla^2 u_s - k^2 u_s = f_s &\qquad \text{in } \ \Omega_s \\ \frac{\partial u_s}{\partial \nu_s} + \alpha u_s = g_s &\qquad \text{on } \ \partial\Omega_s \cap \partial\Omega \\ \frac{\partial u_s}{\partial \nu_s} + iku_s = \lambda_s &\qquad \text{on } \ \Gamma_I \end{aligned}$$

under the double constraint

$$[\frac{\partial u}{\partial \nu_1} + iku] = 0 \quad \text{ and } \quad [\frac{\partial u}{\partial \nu_2} + iku] = 0 \quad \text{ on } \ \Gamma_I$$

where $[p] = p_1 - p_2$ denotes the jump in p on either sides of the interface Γ_I, f_s (resp. g_s) denotes the restriction of the function f (resp. g) in subdomain Ω_s (resp. $\partial\Omega_s \cap \partial\Omega$), and ν_s the external normal vector of Ω_s, for $s = 1, 2$.

This method is called the FETI-H2LM method because it involves two Lagrange multipliers λ_1 and λ_2. It is important to note that the FETI-H2LM method ensures the continuity of the pressure and the continuity of the pressure normal derivative on Γ_I :

$$\begin{aligned} u_1 - u_2 &= 0 \\ \frac{\partial u_1}{\partial \nu_1} + \frac{\partial u_2}{\partial \nu_2} &= 0\ . \end{aligned}$$

These relations ensure that the continuous solution in each subdomain is the restriction of the global continuous solution of the initial problem.

Each local problem is well-posed because of the complex boundary conditions. The variational formulation now leads to the following pseudo-hybrid problem : for $f \in L^2(\Omega)$, for $g \in L^2(\partial\Omega)$, find $u_1 \in H^1(\Omega_1)$, $u_2 \in H^1(\Omega_2)$, $\lambda_1 \in H^{-1/2}(\Gamma_I)$ and $\lambda_2 \in H^{-1/2}(\Gamma_I)$ such that

$$\begin{aligned} \int_{\Omega_1} (\nabla u_1 \nabla v_1 - k^2 u_1 v_1) + ik \int_{\Gamma_I} u_1 v_1 &+ \alpha \int_{\partial\Omega_1 \cap \partial\Omega} u_1 v_1 \\ = \int_{\Omega_1} f_1 v_1 &+ \int_{\partial\Omega_1 \cap \partial\Omega} g_1 v_1 + \int_{\Gamma_I} \lambda_1 v_1 \\ \int_{\Omega_2} (\nabla u_2 \nabla v_2 - k^2 u_2 v_2) + ik \int_{\Gamma_I} u_2 v_2 &+ \alpha \int_{\partial\Omega_2 \cap \partial\Omega} u_2 v_2 \\ = \int_{\Omega_2} f_2 v_2 &+ \int_{\partial\Omega_2 \cap \partial\Omega} g_2 v_2 + \int_{\Gamma_I} \lambda_2 v_2 \\ \int_{\Gamma_I} (\lambda_1 + \lambda_2 - 2iku_2) v &= 0 \\ \int_{\Gamma_I} (\lambda_1 + \lambda_2 - 2iku_1) v &= 0 \end{aligned}$$

and that for all $v_1 \in H^1(\Omega_1)$, $v_2 \in H^1(\Omega_2)$ and $v \in H^{1/2}(\Gamma_I)$. With the same notation, the discretization leads to the linear systems

$$\begin{bmatrix} \tilde{K}_{11} & \tilde{K}_{13} \\ \tilde{K}_{31} & \tilde{K}_{33}^{(1)} + ikM_I \end{bmatrix} \begin{pmatrix} x_1 \\ x_3^{(1)} \end{pmatrix} = \begin{pmatrix} b_1 \\ b_3^{(1)} + \lambda_1 \end{pmatrix}, \tag{2a}$$

$$\begin{bmatrix} \tilde{K}_{22} & \tilde{K}_{23} \\ \tilde{K}_{32} & \tilde{K}_{33}^{(2)} + ikM_I \end{bmatrix} \begin{pmatrix} x_2 \\ x_3^{(2)} \end{pmatrix} = \begin{pmatrix} b_2 \\ b_3^{(2)} + \lambda_2 \end{pmatrix} \tag{2b}$$

with the two constraints on the interface :

$$\lambda_1 + \lambda_2 - 2ikM_I x_3^{(2)} = 0 \tag{3a}$$

$$\lambda_1 + \lambda_2 - 2ikM_I x_3^{(1)} = 0\ . \tag{3b}$$

The matrix M_I is the mass matrix related to the interface Γ_I and thus is symmetric positive-definite. The vector $b_3^{(s)}$ denotes the contribution from subdomain Ω_s to the nodes on the interface and $x_3^{(s)}$ is the solution on the interface computed by solving the local problem in subdomain Ω_s. Clearly, adding the contributions of the elements from Ω_1 and Ω_2 on the interface, gives $\tilde{K}_{33} = \tilde{K}_{33}^{(1)} + \tilde{K}_{33}^{(2)}$ and $b_3 = b_3^{(1)} + b_3^{(2)}$.

Theorem 3.1 *The discrete solution obtained in subdomain Ω_s with the FETI-H2LM method is equal to the restriction of the discrete solution of the initial problem onto Ω_s.*

Proof Starting from Eq. (3), x_3 can be set to $x_3 = x_3^{(1)} = x_3^{(2)}$. Next the assembly of the two previous systems and a simplification give the linear system :

$$\begin{bmatrix} \tilde{K}_{11} & 0 & \tilde{K}_{13} \\ 0 & \tilde{K}_{22} & \tilde{K}_{23} \\ \tilde{K}_{31} & \tilde{K}_{32} & \tilde{K}_{33}^{(1)} + \tilde{K}_{33}^{(2)} \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ b_3^{(1)} + b_3^{(2)} \end{pmatrix}.$$

Recalling the relations $\tilde{K}_{33} = \tilde{K}_{33}^{(1)} + \tilde{K}_{33}^{(2)}$ and $b_3 = b_3^{(1)} + b_3^{(2)}$, this system is exactly the initial discrete problem (1) when the unknowns x in (1) are ordered appropriately. This means that the discrete solution obtained in each subdomain with the FETI-H2LM method corresponds to the restriction in each subdomain of the discrete solution of the original problem.

Sometimes the short notation for the local unknows and right-hand side vectors,

$$u_s = \begin{pmatrix} x_s \\ x_3^{(s)} \end{pmatrix} \quad \text{and} \quad f_s = \begin{pmatrix} b_s \\ b_3^{(s)} \end{pmatrix},$$

will be used. So, the linear systems (2) can be written as

$$\tilde{K}_1 u_1 = f_1 + B_1^t \lambda_1 \quad \text{and} \quad \tilde{K}_2 u_2 = f_2 + B_2^t \lambda_2 \tag{4}$$

with the constraints

$$\lambda_1 + \lambda_2 - 2ikM_I B_2 u_2 = 0 \tag{5a}$$
$$\lambda_1 + \lambda_2 - 2ikM_I B_1 u_1 = 0 \tag{5b}$$

where $\tilde{K}_s$ denotes the new local regularized Helmholtz matrix defined as

$$\tilde{K}_s = K_s - k^2 M_s + \alpha M_R + ikB_s^t M_I B_s$$

and B_s is the discrete restriction from Ω_s on Γ_I. By the elimination of u_s from (4) into (5), the linear system

$$F\lambda = d \tag{6}$$

is obtained where λ is defined by $\lambda = (\lambda_1, \lambda_2)^t$, and F is the non-Hermitian matrix

$$F = \begin{bmatrix} I & I - 2ikM_I B_2 \tilde{K}_2^{-1} B_2^t \\ I - 2ikM_I B_1 \tilde{K}_1^{-1} B_1^t & I \end{bmatrix},$$

and

$$d = \begin{pmatrix} 2ikM_I B_2 \tilde{K}_2^{-1} f_2 \\ 2ikM_I B_1 \tilde{K}_1^{-1} f_1 \end{pmatrix}.$$

This problem is solved by an iterative method and once (λ_1, λ_2) is known, the pressure values u_1 and u_2 can be computed by solving (4).

3.2 Iterative solution of the interface problem

This section is devoted to the iterative solution of the linear system (6) on parallel computers. The reader is referred to (Barrett, Berry, Chan, Demmel, Donato, Dongarra, Eijkhout, Pozo, Romine and van der Vorst 1994, Saad 1996, Dongarra, Duff, Sorensen and van der Vorst 1998) for reviews on iterative methods. In this paper, unpreconditioned GMRES(m) and BiCGStab(ℓ) are considered, so the only operation for F required is a matrix-vector product.

The conjugate gradient method is designed for Hermitian positive definite matrices and the work per iteration is usually dominated by the matrix-vector product with F. The storage of five vectors of the dimension of λ is required. The method is optimal since it minimizes the error in the F-norm and has a smooth convergence behaviour. For non-Hermitian matrices, other methods must be used. The GMRES(m) method, proposed by Saad and Schultz (Saad and Schultz 1986), keeps the property of optimal and smooth convergence behaviour, but the memory requirements can be large, since the basis vectors of a large Krylov space should be stored (cf. Algorithm 1). One step of the GMRES(m), as defined in Algorithm 1, by 3.1–3.7 requires one matrix-vector product. The quantity TOL is the residual tolerance used for the stopping criterion.

Algorithm 1 (GMRES(m))

```
1. Compute the residual $r = d - F\lambda_0$
   Compute : $\beta = ||r||$
2. First basis vector : $v_1 = r/\beta$
3. For $j = 1, \ldots, m$.
    3.1. Compute $w_j = Fv_j$
    3.2. Compute Gram-Schmidt coefficients : $h_{ij} = v_i^H w_j$, $i = 1, \ldots, j$
    3.3. Gram-Schmidt orthogonalization : $w_j = w_j - \sum_{i=1}^{j} h_{ij} v_i$
    3.4. Normalization : $v_{j+1} = w_j / h_{j+1,j}$ with $h_{j+1,j} = ||w_j||$
    3.5. Let $H_j = [h_{il}]_{(i,j)=(1,1)}^{(j+1,j)}$ with $h_{il} = 0$ for $i > l + 1$
    3.6. Solve the least squares problem $H_j z_j = \beta e_1$
    3.7. Update the solution $\lambda = \lambda_0 + [v_1, \ldots, v_j] z_j$
4. Compute $r = d - F\lambda$
5. If $||r|| > \mathrm{TOL}$, set $\lambda_0 = \lambda$, and go to step 1.
```

Methods based on biconjugate gradients require less memory, but the convergence behaviour is not optimal and can be very irregular. Among this type of methods, the BiCGStab(ℓ) method is selected as a compromise between smooth convergence and small storage requirements. It combines ℓ steps of the biconjugate gradient method (BiCG) with ℓ steps of GMRES (cf. Algorithm 2). In this way, the method is able to keep the memory requirements low, but at the same time uses the stabilizing effect of GMRES. The implementation of BiCGStab(ℓ) is rather technical and the reader is referred to Fokkema (Fokkema 1996) for software details. One step of the BiCGStab(ℓ) as defined in Algorithm 2 by 2.2.1–2.2.8 requires two matrix-vector product.

Algorithm 2 (BiCGStab(ℓ))

```
1. Compute the residual $\tilde{r}_0 = r_0 = d - F\lambda_0$
   Set $u_0 = 0$, $\alpha = \rho_0 = \omega = 1$.
```

```
2. While $||r_0|| > TOL$ do
  2.1. $\rho_0 = -\omega\rho_0$
  2.2. For $j = 0, \ldots, \ell - 1$
      2.2.1. $\rho_1 = \tilde{r}_0^H r_j$, $\beta = \alpha(\rho_1/\rho_0)$
      2.2.2. $\rho_0 = \rho_1$.
      2.2.3. Update search vectors $[u_0, \ldots, u_j] = [r_0, \ldots, r_j] - \beta[u_0, \ldots, u_j]$
      2.2.4. Compute the new search vector $u_{j+1} = Fu_j$.
      2.2.5. Compute $\sigma = \tilde{r}_0^H u_{j+1}$, $\alpha = \rho_1/\sigma$.
      2.2.6. Update the solution $\lambda = \lambda + \alpha u_0$
      2.2.7. Update the residuals $[r_0, \ldots, r_j] = [r_0, \ldots, r_j] - \alpha[u_1, \ldots, u_{j+1}]$
      2.2.8. Compute the new residual vector $r_{j+1} = Fr_j$.
  2.3. Compute $Z = [r_1, \ldots, r_\ell]^H [r_1, \ldots, r_\ell]$
  2.4. Solve the problem $Zy = [r_1, \ldots, r_\ell]^H r_0$
  2.5. Update the solution $x = x + [r_0, \ldots, r_{\ell-1}]y$
  2.6. Update the residual vector $r_0 = r_0 - [r_1, \ldots, r_\ell]y$
  2.7. Update the search vector $u_0 = u_0 - [u_1, \ldots, u_\ell]y$
```

The parallel solution of the linear system ($F\lambda = d$) takes place as follows. The vectors λ, d as well as the iteration vectors are distributed in the same way among the processors of the distributed computer. The computation of the matrix-vector product, in parallel, is discussed in 4. The operations on the vectors of large dimension are all carried out in parallel. This does not require communication between processors for vector updates ($y \leftarrow y + \alpha x$) since this operation can be carried out for each component of y independently, but communication is required for the inner product $\zeta = x^H y$. The other operations are carried out simultaneously without communication on each processor. The reader is referred to (Barrett et al. 1994, Dongarra et al. 1998) for implementation details.

4 Implementation aspects

4.1 Decomposition into subdomains

In a finite element method, the elements are connected via their faces; these are lines in a 2-D model and surfaces in 3-D. The elements of the global domain are decomposed into N_s domains by numbering or colouring all elements. All elements with the same number or colour form a subdomain. Subdomains are sets of elements, so the interface consists of a set of faces. The interface is defined by the faces connected to elements with a different subdomain number. The domain decomposition method is by nature parallel : the solution can be computed independently for each subdomain, while the interface equation connects all the independent subdomains into a global problem. Because of this independence, each domain can be allocated to a single processor of the parallel system, for example. Operations related to a single subdomain take place without communication. The solution of the interface problem, however, requires communication since it connects all the subdomains.

The decomposition should be done in such a way that the work to solve the N_s local problems is equal in order to obtain a good load balancing among the subdomains and so that the number of interface nodes is small in order to have a small interface problem. In many cases, this decomposition can be done by hand, but for practical applications, it is often a very difficult and tedious task. The decomposition into subdomains is intensively studied using graph theory and the reader is referred to the references in (Karypis and Kumar 1997). In this paper, the meshes of the test problems were decomposed by hand so that each subdomain has almost the same number of unknowns.

4.2 Matrix-vector product by the interface operator

In 3.1, the FETI-H2LM method for two subdomains has been discussed. For each interface degree of freedom (dof), there are two Lagrange multipliers, one for each subdomain. The interface boundary conditions make the link between the multipliers on both sides of the interface. This concept is still valid when more than two subdomains are present as long as the interface nodes only connect two neighbouring subdomains. It may happen, however, that three or more subdomains share a node at the interface as illustrated in Figure 1. Subdomain 1 is connected to subdomain 2 via common faces and similarly to subdomain 3, but not to subdomain 4, since subdomains 1 and 4 only share a common node. The Lagrange multipliers for nodes that connect more than two subdomains are duplicated in order to be able to use the formulation for two subdomains. The consequence is that, for each subdomain, the number of Lagrange multipliers is, in general, larger than or equal to the number of unknowns on the interface. In this situation, subdomain 1 has at node c, a multiplier λ_2 that is connected to domain 2 and a multiplier λ_3 connected to domain 3. In this case, both λ_2 and λ_3 should be added together to the same row in the right-hand side of (4) for the solution of the local problem. The relation between the Lagrange multipliers λ_s and the local pressure unknowns u_s is given by the restriction matrix B_s, i.e. $\lambda_s = B_s u_s$. When the interface connects two subdomains, B_s just copies the corresponding coefficients from u_s into λ_s. When more than two subdomains are involved, then a pressure unknown can be copied to more than one Lagrange multiplier. The inverse operation $u_s = B_s^t \lambda_s$, is very similar. When the interface involves two subdomains, λ_s's are copied to u_s's. When more subdomains are involved, the λ_s's are added together into the corresponding u_s.

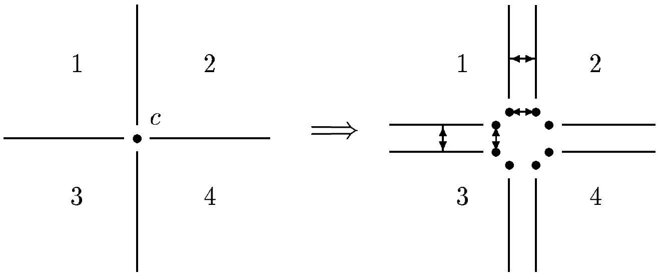


Figure 1: Illustration of four subdomains with a common node. The arrows denote the action of the interface boundary conditions for subdomain 1.

Here is the algorithm for the computation of the matrix-vector product with the interface matrix.

Algorithm 3 ($\mu = F\lambda$)

```
1. Set the right-hand side $f_s = B_s^t \lambda_s$ for $s = 1, \ldots, N_s$.
   This operation adds all $\lambda_s$'s together that correspond to the same pressure unknown.
2. Solve the local problem $\tilde{K}_s u_s = f_s$ where $s = 1, \ldots, N_s$,
3. Restrict the solution to the interface $\lambda_s' = B_s u_s$.
   This operation copies the pressure unknowns to the Lagrange multipliers.
4. Put $\lambda_s$ in $\lambda_s''$. Exchange each component of $\lambda_s''$ with the corresponding neighbouring
   subdomain.
5. Compute the sum $\lambda_s''' = \lambda_s + \lambda_s''$
6. Add together and apply the regularizing mass matrix : $\mu = \lambda_s''' - 2ikM_I\lambda_s'$
```

The work in Algorithm 3 is dominated by two operations.

- First, the solution of the local problem by a (sequential) sparse direct method in Step 2. The computational cost depends on the number of unknowns in the subdomain and the connectivity of the nodes, i.e. the geometry of the mesh. Therefore, the global domain is decomposed in such a way that the subdomains have (almost) the same number of unknowns. Ideally, the decomposition should happen such that the matrix factorization is equally expensive for all subdomains.

- Second, the communication cost for the iterative methods increases with a larger number of interface variables. This implies that a small interface is preferred.

The FETI-H2LM method was integrated within the acoustic simulation package SYSNOISE (SYSNOISE Rev 5.4 1999). The MPI library (Forum 1994) was used for communication between the subdomains. For the solution of the interface system, GMRES(m) and BiCGStab(2) were used. These use matrix-vector products by the interface matrix. No preconditioning is employed.

The computation of vector inner products is performed by global communication commands from the MPI library (MPI_Allreduce). The current implementation does not overlap communication and computation. Step 4 in Algorithm 3 requires communication between neighbouring subdomains. Non-blocking communication was used for the computation of the mean of λ. The communication consists of a sequence of two-processor exchanges, that represent two neighbouring subdomains. These are the MPI commands MPI_Isend, and MPI_Irecv. The other operations in Algorithm 3 are fully parallel without communication.

The local assembled finite element matrix $\tilde{K}_s$ was factorized by the SYSNOISE built-in direct solver, which is an LDL^t factorization. Since each evaluation with F by Algorithm 3 requires the solution of a linear system with $\tilde{K}_s$, it is advantageous to factorize once and perform just the forward and backward substitutions in Step 2 of Algorithm 3.

4.3 Theoretical computational cost

In this section, the operation count is analyzed for a (simple) model problem with two subdomains. Figure 2 shows a rectangular mesh with $2n$ elements in vertical direction and m elements horizontally. The entire domain is decomposed into two subdomains Ω_1 and Ω_2 of $n \times m$ elements each, with the interface Γ_I of $p = m + 1$ unknowns. This section presents theoretical results for the operation counts for a parallel direct method and the FETI-H2LM method. We assume that $n, m \gg 1$.

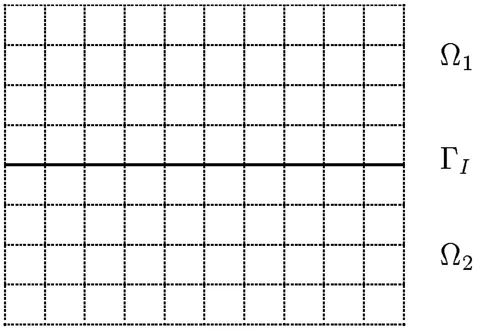


Figure 2: Decomposition of an $m \times 2n$ rectangular mesh of elements. (Here $n = 4$ and $m = 10$.)

Following the conventions on the ordering of the unknowns, the assembled linear system has the block form

$$\begin{bmatrix} \tilde{K}_{11} & 0 & \tilde{K}_{13} \\ 0 & \tilde{K}_{22} & \tilde{K}_{23} \\ \tilde{K}_{31} & \tilde{K}_{32} & \tilde{K}_{33} \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix} . \tag{7}$$

Assuming that $\tilde{K}_{ss}$ can be factorized into $\tilde{K}_{ss} = L_{ss}L_{ss}^T$ with L_{ss} lower triangular, a linear system can be solved using the block factorization

$$\begin{bmatrix} L_{11} & 0 & 0 \\ 0 & L_{22} & 0 \\ \tilde{K}_{31}L_{11}^{-T} & \tilde{K}_{32}L_{22}^{-T} & I \end{bmatrix} \begin{bmatrix} L_{11}^T & 0 & L_{11}^{-1}\tilde{K}_{13} \\ 0 & L_{22}^T & L_{22}^{-1}\tilde{K}_{23} \\ 0 & 0 & S \end{bmatrix} .$$

Note that $\tilde{K}_{3s}L_{ss}^{-T} = (L_{ss}^{-1}\tilde{K}_{s3})^T$, so the Schur complement

$$S = \tilde{K}_{33} - \sum_{s=1,2} L_{s3}^T L_{s3} \quad \text{with} \quad L_{s3} = L_{ss}^{-1}\tilde{K}_{s3} .$$

Let the unknowns in each subdomain be ordered columnwise. The matrix $\tilde{K}_{ss}$ has $N = np$ rows and columns and is a band matrix with bandwidth $\mathrm{BW} = n$. The matrix factorization of $\tilde{K}_{ss}$ without pivoting then requires approximately $2 \cdot \mathrm{BW}^2 \cdot N = 2n^3p$ operations. Computing $L_{s3} = L_{ss}^{-1}\tilde{K}_{s3}$ requires p forward substitutions of $\mathrm{BW} \cdot N$ operations each. This gives a total of n^2p^2 operations. Since $\tilde{K}_{s3}$ is sparse, the computation of L_{s3} can be organized in a more efficient way so that the total cost is $\alpha n^2 p^2$ with $0 < \alpha \leq 1$ where α is a reduction factor that depends on the sparsity of $\tilde{K}_{s3}$. We can show that for this example, $\alpha = 0.5$. Once L_{s3} is known, the term $L_{s3}^T L_{s3}$ in S can be computed. Since S is symmetric, it is sufficient to form only the upper triangular part. When L_{s3} is dense, this work is of the order $2 \cdot N \cdot \frac{1}{2}p^2 = np^3$. Taking into account the sparsity of L_{s3}, we can show that the cost of this operation can be reduced to $\frac{1}{3}np^3$ for this example. The addition of the terms in S is of the order of p^2 operations and the factorization of S requires about p^3 operations. When m and n are large, these are negligible to the other operations. When two processors are used, the factorizations of $\tilde{K}_{11}$ and $\tilde{K}_{22}$ takes place in parallel, as well as the computation of L_{13} and L_{23}. Roughly speaking, the cost on each processor for the parallel direct method described here is of the order of

$$2n^3p + \frac{1}{2}n^2p^2 + \frac{1}{3}np^3 \ .$$

For the FETI-H2LM method, the local matrices $\tilde{K}_s$ contain the assembled matrix for the elements in Ω_s including Γ_I. This implies that $\tilde{K}_s$ has $N = (n+1)p$ rows and columns. If the unknowns are numbered columnwise, the bandwidth is $\mathrm{BW} = n+1$. The number of operations for the factorization of $\tilde{K}_s$ is $2 \cdot \mathrm{BW}^2 \cdot N = 2(n+1)^3p$. The forward and backward substitutions with $\tilde{K}_s$ cost about $2 \cdot \mathrm{BW} \cdot N = 2(n+1)^2p$ operations. The most expensive operation in the iterative solvers is the matrix-vector product with F, which is in turn dominated by the local solve with $\tilde{K}_s$. On each iteration, each processor performs one forward and one backward substitution, i.e. requires $2 \cdot \mathrm{BW} \cdot N = 2(n+1)^2p$ operations. For q iterations, the total cost (including factorization) is of the order of

$$2(n+1)^3p + 2q(n+1)^2p \approx 2n^3p + 2qn^2p \ .$$

Compared with the direct parallel method, the operation count of the FETI-H2LM method is smaller than the operation count of the direct method when

$$\begin{aligned} 2n^3p + 2qn^2p &\leq 2n^3p + \frac{1}{2}n^2p^2 + \frac{1}{3}np^3 \\ q &\leq \left(\frac{1}{4} + \frac{1}{3}\frac{p}{n}\right)p \ . \end{aligned}$$

In words, the FETI-H2LM method requires less floating point operations than the parallel direct method described here, when the number of iterations is much smaller than the number of interface variables.

5 Numerical results

The numerical examples have been selected in order to demonstrate the current capabilities of the presented method. The first example is related to a car compartment and the second is dealing with an exhaust system. These problems were solved using the SYSNOISE software tool for vibro-acoustic simulation (SYSNOISE Rev 5.4 1999), on a four-processor SGI Origin 200, using the MPI library.

5.1 Car compartment

The first example is related to a car compartment. The main objective of this evaluation is the synthesis of the frequency response function, at the driver's and passenger's ears, subjected to some velocity boundary conditions along the firewall. This example is representative of a wider class of problems where the acoustic response within a cavity is evaluated as induced by vibrating panels.

The evaluation of the acoustic response is performed using a three-dimensional finite element model as presented in Fig. 3. The discrete model involves 6448 hexaedral elements, 544 pentaedral elements, and 8417 nodes. Neumann boundary conditions are considered along the firewall, and Robin boundary conditions are considered along the ceiling. Homogeneous Neumann boundary conditions are considered elsewhere as presented in the longitudinal cut of the car Fig. 3. The two mesh partitions represented Fig. 4 have been used for the numerical simulation. The first consists of two subdomains ($N_s = 2$) with 4429 degrees of freedom (dofs) each and the interface has 441 dofs. The second partition in four subdomains ($N_s = 4$) consists of two subdomains with 2443 dofs and two with 2229 dofs ; the interface has 955 dofs. The size of the interface problem for the FETI-H2LM method is the double of the size of the interface, because of the use of two Lagrange multipliers. This leads to an interface problem of size 882 for the first partition and to one of size 1910 for the second.

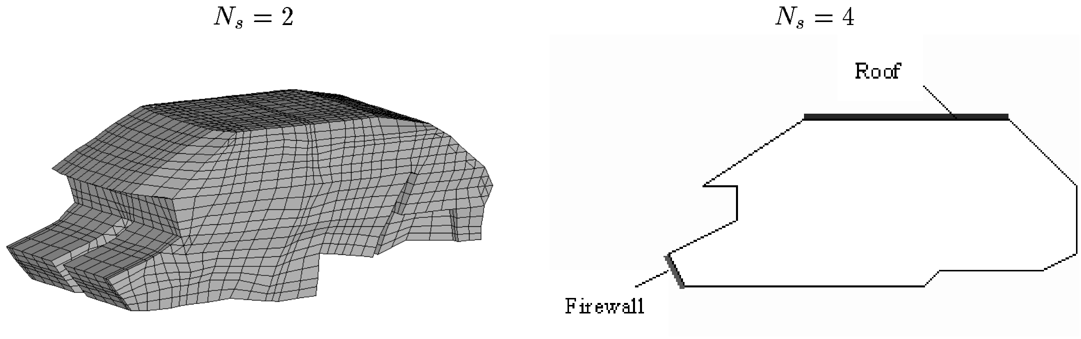
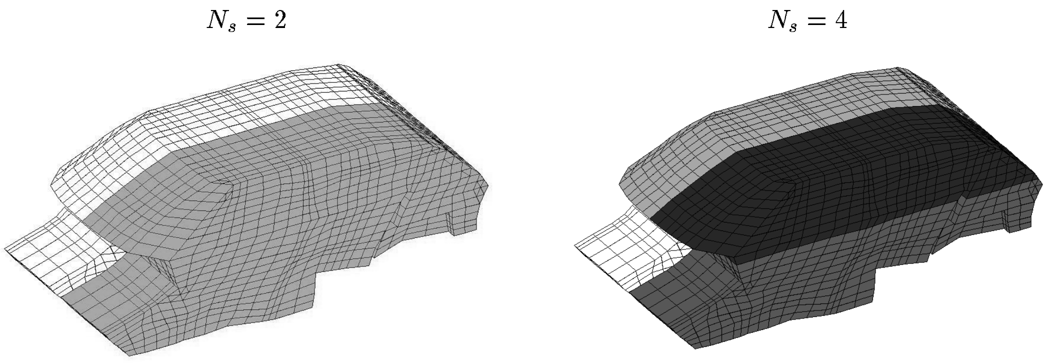


Figure 3: Car compartment — mesh and boundary conditions

The convergence curves obtained for the two partitions for 100 Hz are presented in Fig. 5. This figure shows the influence of the choice of iterative method. Three algorithms have been used, namely GMRES(20), GMRES(50), and BiCGStab(2). The curves show the scaled interface residual $||F\lambda - d||/||d||$ versus the iteration number.

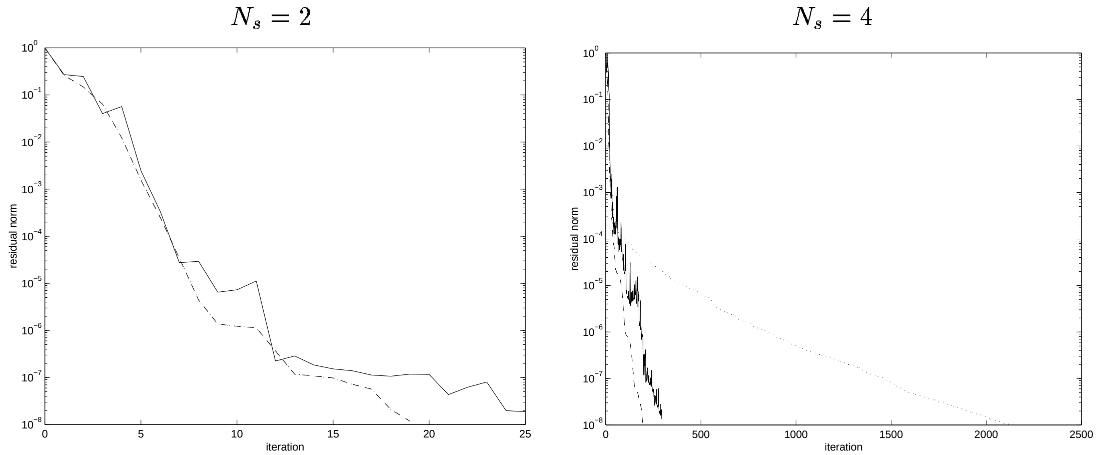
For all cases, GMRES(50) is the fastest in terms of iterations. Note that the figures do not represent the cost per iteration. Indeed, GMRES(m) requires more (parallel) inner products than BiCGStab(2), which can lead to an important overhead. But, GMRES(m) requires only one matrix-vector product per iteration and BiCGStab(2) two. Another difference between GMRES(m) and BiCGStab(2) is the convergence behaviour : GMRES(m) converges smoothly, while BiCGStab(2) has a somewhat irregular behaviour.

It can be seen that the decomposition into four domains leads to a more difficult problem, since each of the methods require more iterations. This is very pronounced for



$N_s = 2$ $N_s = 4$

Figure 4: Car compartment — domain decompositions



$N_s = 2$ $N_s = 4$

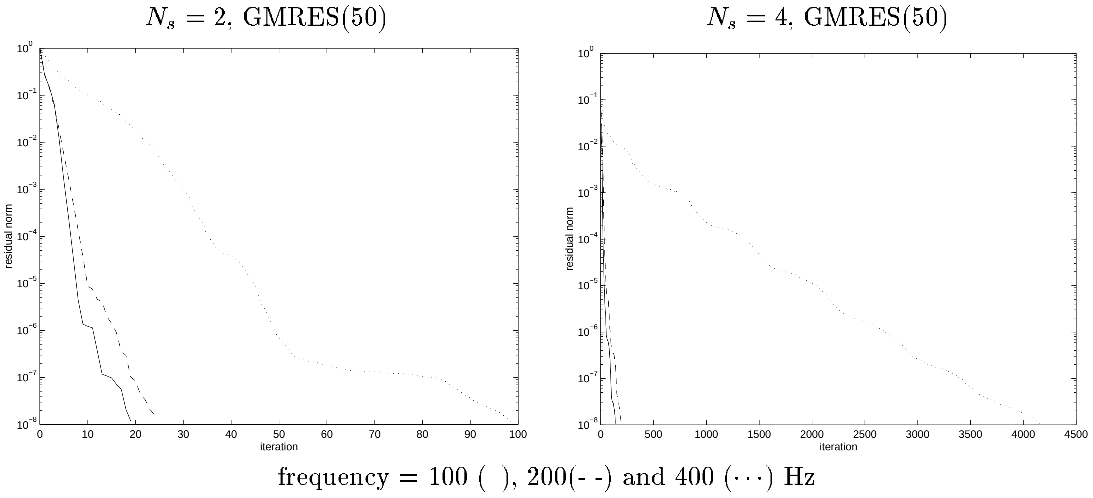
method = GMRES(20) (···), GMRES(50) (- -) and BiCGStab(2) (—)

Figure 5: Car compartment — convergence history for 100 Hz

GMRES(20), which requires more than 2000 iterations in the case of four subdomains and converges very slowly, compared to the case of two subdomains. The difference in convergence behaviour between GMRES(20) and GMRES(50) shows how important the number of iteration vectors is. Clearly, the robustness of GMRES(m) can only be guaranteed when the number of iteration vectors m is large enough.

The GMRES(50) and the BiCGStab(2) have been used for computing the acoustic response for the frequencies 100, 200 and 400 Hz. Selecting the total length of the cavity as the reference length (a = 286 cm), the corresponding reduced frequencies (ka) are 1.848, 3.696 and 7.392.

The dependency on the number of subdomains and the frequency for GMRES(50) are shown in Fig. 6. It is clear that when $N_s = 4$, the number of iteration vectors is too small for fast convergence. The results for BiCGStab(2) are presented Fig. 7. The dependency on frequency and number of subdomains is similar to GMRES(m). The GMRES(20) method performs very badly. Since for $N_s = 2$, the number of interface nodes ($p = 441$) is about 10 to 20 times the number of matrix-vector products of the iterative methods for 100 and 200Hz, the FETI-H2LM method is expected to be cheaper in floating point operations than the parallel direct method described in §4.3. This is not true for 400Hz.



$N_s = 2$, GMRES(50)

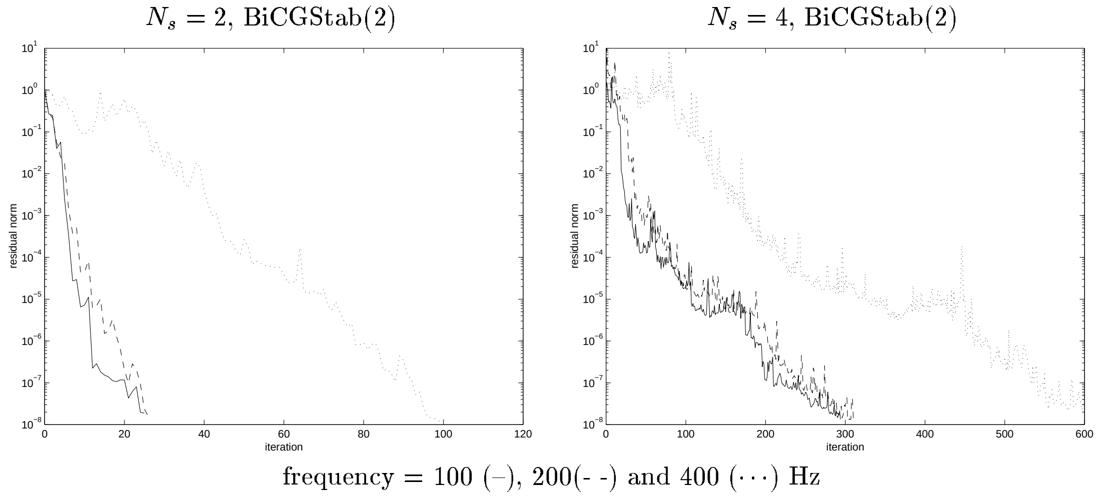
$N_s = 4$, GMRES(50)

frequency = 100 (–), 200(- -) and 400 (···) Hz

Figure 6: Car compartment — convergence history

5.2 Exhaust system

The second example is related to an exhaust line of a Jaguar car. For such a silencer, the main objective is the evaluation of acoustic transmission properties. The length of the tube, the location of expansion chambers, the particular geometry of these chambers and the use of absorbent materials and/or perforated facings are some factors which could affect these transmission properties. Useful characteristics of such a system are the transmission loss or the insertion loss.


$N_s = 2$, BiCGStab(2)

$N_s = 4$, BiCGStab(2)

frequency = 100 (–), 200(- -) and 400 (···) Hz

Figure 7: Car compartment — convergence history

The finite element model presented here involves 39254 hexaedral elements and 46966 nodes. Dirichlet or Neumann boundary conditions along inlet and outlet sections are imposed ; homogeneous Neumann conditions are set on the other boundaries. The mesh was partitioned into two subdomains ($N_s = 2$), as shown Fig. 8. Each subdomain has 23483 unknowns and the interface has 347 unknowns. The associated interface system is of order 694.

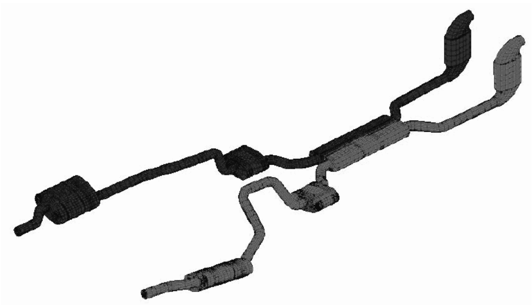
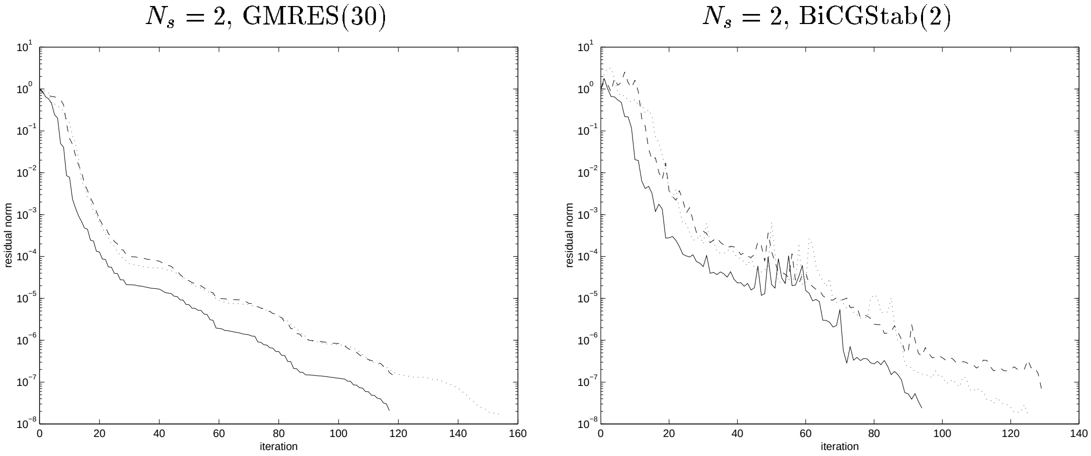


Figure 8: Exhaust system — domain decomposition ($N_s = 2$)

The computations were performed for frequencies 100, 200 and 500 Hz. Convergence curves for GMRES(30) and BiCGStab(2) of the interface residual $||F\lambda - d||/||d||$ are presented in Figure 9. In contrast with the previous example of the car compartment, the dependency on the frequency is less pronounced. Obviously, the geometry of the problem and the decomposition reduces the dependency on the frequency. Since the number of interface nodes ($p = 347$) is about three times the number of matrix-vector products for GMRES(30) and of the same order for BiCGStab(2), it is expected that

FETI-H2LM will not be more efficient than a direct parallel method.


$N_s = 2$, GMRES(30)

$N_s = 2$, BiCGStab(2)

frequency = 100 (–), 200 (- -) and 500 ($\cdots$) Hz

Figure 9: Exhaust system — convergence history

6 Conclusions

In this paper, a brief overview of a one-level Finite Element Tearing and Interconnecting method for the Helmholtz equation has been presented. The performance of this method has been studied for two representative three-dimensional applications originating from the automotive industry. Two iterative solvers for the interface problem have been used. The following conclusions can be drawn. First, the behaviour of GMRES(m) strongly depends on the number of iteration vectors m. The number of iteration vectors should be chosen quite large in order to obtain acceptable convergence speed. For the two examples tested, more than 25 vectors are recommended. This is feasible, because those vectors are defined on the interface and their storage is small compared to the storage of the local regularized Helmholtz matrix in each subdomain. Second, the BiCGStab(2) method does the job very well in terms of the number of iterations. Its storage cost is relatively low, since only nine vectors are required, but two matrix-vector products are involved at each iteration. Furthermore, the convergence behaviour is more irregular than GMRES(m). Third, increasing the number of subdomains leads to a more difficult problem and the solution becomes more expensive for higher frequencies, but this property depends on the geometry.

The three-dimensional applications of the FETI-H2LM method without preconditioning on a car compartment and on an exhaust system depend on the frequency and the number and the shape of the subdomains in a similar way as two-dimensional applications. This is discouraging since domain decomposition is potentially the method of choice for higher frequencies where fine meshes are required. However, this problem may be solved with a global preconditioner based on Krylov spaces and the method can be derived as a two-level Finite Element Tearing and Interconnecting method as introduced in (de La Bourdonnaye et al. 1998).

Acknowledgements

The authors acknowledge partial support by the European Commission under Grant ESPRIT-25009. We are grateful to LMS International, partner in this project, for providing the test cases and equipment for running the numerical examples. We also thank Jacko Koster and Jennifer Scott (Rutherford Appleton Laboratory, United Kingdom), Yvon Maday (University of Paris 6, France) and François-Xavier Roux (ONERA, France) for useful comments that improved the quality of the paper.

References

Barrett, R., Berry, M., Chan, T., Demmel, J., Donato, J., Dongarra, J., Eijkhout, V., Pozo, R., Romine, C. and van der Vorst, H. (1994), *Templates for the solution of linear systems: building blocks for iterative methods*, SIAM, Philadelphia, PA, USA.

Chevalier, P. and Nataf, F. (1998), 'Symmetrized method with optimized second-order conditions for the Helmholtz equation', *Contemporary Mathematics* **218**, 400–407.

Collino, F. (1993), Conditions absorbantes d'ordre élevé pour les équations de Maxwell dans des domaines rectangulaires, Research Report 19-91, INRIA.

de La Bourdonnaye, A., Farhat, C., Macedo, A., Magoulès, F. and Roux, F. (1998), 'A non-overlapping domain decomposition method for the exterior Helmholtz problem', *Contemporary Mathematics* **218**, 42–66.

Després, B. (1990), 'Décomposition de domaine et problème de Helmholtz', *Compte Rendu de l'Académie des Sciences, France* **I**(311), 313–316.

Dongarra, J. J., Duff, I. S., Sorensen, D. C. and van der Vorst, H. A. (1998), *Numerical Linear Algebra for High-Performance Computers*, Software - Environments - Tools, SIAM, Philadelphia, PA 19104-2688, USA.

Duff, I. (1998), Direct methods, Technical report RAL-TR-1998-054, Rutherford Appleton Laboratory, Chilton, Didcot, OX11 0QX UK.

Farhat, C. and Roux, F. (1994), 'Implicit parallel processing in structural mechanics', *IACM, Computational Mechanics Advances* **2**(1), 121–220. North Holland.

Farhat, C. and Roux, F.-X. (1992), 'An unconventional domain decomposition method for an efficient parallel solution of large-scale finite element systems', *SIAM J. Sci. Statist. Comput.* **13**, 379–396.

Farhat, C., Macedo, A. and Lesoinne, M. (n.d.*a*), 'A two-level domain decomposition method for the iterative solution of high frequency exterior Helmholtz problems', *Numerische Mathematik.* (submitted for publication).

Farhat, C., Macedo, A., Lesoinne, M., Roux, F.-X., Magoulès, F. and Bourdonnaye, A. d. l. (n.d.*b*), 'Two-level domain decomposition methods with Lagrange multipliers for the fast iterative solution of acoustic scattering problems', *Computer Methods in Applied Mechanics and Engineering.* (submitted for publication).

Fokkema, D. (1996), Enhanced implementation of BiCGSab(ℓ) for solving linear systems of equations, Preprint 976, Universiteit Utrecht, Department of Mathematics.

Forum, M. P. I. (1994), 'MPI : a message-passing interface standard', *Journal of Supercomputer Applications and High Performance Computing.*

Karypis, G. and Kumar, V. (1997), METIS : a software package for partitioning unstructured graphs, partitioning meshes, and computing fill-reducing orderings of sparse matrices, Available via `http://www.cs.umn.edu/~karypis`.

Lions, J.-L. and Dautray, R. (1985), *Analyse mathémathique et calcul numérique pour les sciences et techniques*, Tome 1, Masson, Paris.

Magoulès, F., Roux, F. and de La Bourdonnaye, A. (1998*a*), 'Méthode de décomposition de domaines pour des problèmes hyperboliques', *Calculateurs Parallèles Réseaux et Systèmes répartis* **10**(4), 353–361.

Magoulès, F., Roux, F., Coyette, J. and Lecomte, C. (1998*b*), Numerical treatment of internal acoustic problems by substructuring methods, Proceedings of the International Conference on Advanced Computational Methods in Engineering. Ghent, Belgium.

Pierce, A. D. (1981), *Acoustics - An introduction to its physical principles and applications*, Mac Graw Hill, New York.

Roux, F. (1995), 'Méthodes de décomposition de domaine pour des problèmes elliptiques', *Calculateurs Parallèles* **7**(3), 237–253.

Saad, Y. (1990), SPARSKIT : a basic tool kit for sparse matrix computations, Technical report TR-90-20, Research Institute for Advanced Computer Science, NASA Ames Research Center, Moffet Field, CA.

Saad, Y. (1996), *Iterative methods for linear systems*, PWS Publishing, Boston.

Saad, Y. and Schultz, M. (1986), 'GMRES : a generalized minimal residual algorithm for solving non-symmetric linear systems', *SIAM J. Sci. Statist. Comput.* **7**, 856–869.

SYSNOISE Rev 5.4 (1999), A software tool for vibro-acoustic simulation, LMS International, Interleuvenlaan 70, 3001 Leuven, Belgium.

Tallec, P. l. (1994), 'Domain decomposition methods in computational mechanics', *I.A.C.M., Computational Mechanics Advances* **1**(2), 121–220. North Holland.