

FETI-DP for 3D magnetostatic problems discretized by edge elements

Mohamed I. Ghenai^{1,2,3}, Ronan Perrussel², Olivier Chadebec¹, Frédéric Vi³,
Jean-Michel Guichon¹, Gérard Meunier¹, and Jonathan Siau³

¹Univ. Grenoble Alpes, CNRS, Grenoble INP, G2ELab, F-38000, Grenoble, France

²LAPLACE, Université de Toulouse, CNRS, INPT, UPS, Toulouse, France

³Altair Engineering, Meylan, France

Domain decomposition is a numerical method to solve linear and nonlinear problems. It enables fast and parallel execution of simulations. The growing size of electromagnetic simulations raises a lot of interest in developing such methods. Their adaptation in magnetostatics means handling the case of vector potential formulation with its difficulties, mainly resulted from the necessity of gauge condition. This paper provides an approach to overcome this issue and presents test cases for the use of domain decomposition with edge elements.

Index Terms—Domain decomposition, FETI method, Vector potential formulation, Regularization, Parallel computing.

I. INTRODUCTION

MAGNETOSTATIC modeling relies on two types of formulations: scalar potential and vector potential. The latter offers robust solving capabilities but requires a gauge condition to ensure solution unicity. This becomes particularly challenging when employing domain decomposition due to the decomposed nature of the global problem. Domain decomposition, by definition, aims to break down a global problem into smaller ones that can be solved in parallel. Typically, a gauge tree or auto-gauging is employed during solving, but with domain decomposition, local matrix manipulation hinders the use of these methods. In this study, we adopt a different approach based on regularization techniques that can be applied to local problems, enabling the use of the FETI-DP algorithm. The paper presents a test case and results pertaining to conditioning and the number of iterations.

II. PROBLEM DESCRIPTION

A 3D magnetostatic problem using a vector potential formulation will be considered in the remainder of this paper [1] and we will therefore use the following equations

$$\mathbf{b} = \text{curl } \mathbf{a}, \quad \mathbf{b} = \mu \mathbf{h}, \quad \text{curl } \mathbf{h} = \mathbf{j}, \quad (1)$$

where \mathbf{a} is the vector potential, \mathbf{b} the magnetic induction, \mathbf{h} the magnetic field strength, \mathbf{j} the given current density and μ the magnetic permeability. The potential \mathbf{a} is then given by

$$\text{curl}(\mu^{-1} \text{curl } \mathbf{a}) = \mathbf{j}. \quad (2)$$

On a domain Ω and using suited boundary conditions, the variational formulation can be written as

$$(\mu^{-1} \text{curl } \mathbf{a}, \text{curl } \mathbf{a}') = (\mathbf{j}, \mathbf{a}'), \quad \forall \mathbf{a}' \in H_0(\text{curl}; \Omega), \quad (3)$$

where (\cdot, \cdot) denotes the classical L^2 scalar product and $H_0(\text{curl}; \Omega)$ is the space of vector fields whose curl is square-integrable. Note that the vector potential \mathbf{a} is determined up to the gradient of an arbitrary scalar function. A gauge condition is then needed to ensure the unicity of this potential.

Using the edge element method, for a domain Ω with a linear magnet as source, linear ferromagnetic materials and the device surrounded by air, the system to be solved is written

$$Ku = f, \quad (4)$$

where

- K is the finite element matrix of the magnetic vector potential formulation,
- u contains the field circulations of the vector potential on each edge of the mesh,
- f is the source vector.

The finite element matrix contains several null pivots. When solving, a gauge tree or an autogauging method is used to help detect those null pivots. The use of a direct solver to solve the problem is then possible [2].

III. DOMAIN DECOMPOSITION

For this study, we consider non-overlapping domain decomposition methods. They are characterized by the absence of a shared surface (2D) or volume (3D) connecting neighboring subdomains [3]. This choice was motivated by the parallel computing capabilities that it offers.

Here we will focus on the FETI-DP approach (Finite Element Tearing and Interconnecting - Dual Primal). This method is an evolution of the FETI-1 method introduced by Farhat and Roux in 1991 [4]. In FETI-1, the domain is decomposed into subdomains and the continuity on the interface between subdomains is enforced by Lagrange multipliers. Thus, the local subdomain problems are solved independently on a processor by a direct solver, while an iterative method enables to solve a global problem for the multipliers condensed on the interface. FETI-DP methods add a coarse problem with some *primal* variables [5], *i.e.* a few continuity constraints are always satisfied and does not require multipliers, the *dual* variables. An iterative method is used to solve the corresponding problem. The problem to be solved contains then both Lagrange multipliers and the primal variables. These primal variables are typically defined on some cross points so that for a subdomain

s , K_s , u_s and f_s can be partitioned into (interior + remaining boundary) variables denoted by r and coarse primal variables denoted by c :

$$K_s = \begin{bmatrix} K_{s,rr} & K_{s,rc} \\ K_{s,rc}^t & K_{s,cc} \end{bmatrix}, \quad u_s = \begin{bmatrix} u_{s,r} \\ B_{s,c} u_c \end{bmatrix}, \quad f_s = \begin{bmatrix} f_{s,r} \\ f_{s,c} \end{bmatrix}. \quad (5)$$

The problem to solve can then be written as

$$\begin{pmatrix} F_{rr} & F_{rc} \\ F_{rc}^t & -K_{cc} \end{pmatrix} \begin{pmatrix} \lambda \\ u_c \end{pmatrix} = \begin{pmatrix} d_r \\ f_c \end{pmatrix} \quad (6)$$

with F_{rr} , F_{rc} , K_{cc} , d_r et f_c defined as :

$$F_{rr} = \sum_{s=1}^{s=N_{sd}} B_{s,r} K_{s,rr}^{-1} B_{s,r}^t, \quad (7)$$

$$F_{rc} = \sum_{s=1}^{s=N_{sd}} B_{s,r} K_{s,rr}^{-1} K_{s,rc} B_{s,c}, \quad (8)$$

$$K_{cc} = \sum_{s=1}^{s=N_{sd}} B_{s,c}^t K_{s,cc} B_{s,c} - \sum_{s=1}^{s=N_{sd}} (K_{s,rc} B_{s,c})^T K_{s,rr}^{-1} (K_{s,rc} B_{s,c}), \quad (9)$$

$$d_r = \sum_{s=1}^{s=N_{sd}} B_{s,r} K_{s,rr}^{-1} f_{s,r} \quad (10)$$

$$f_c = \sum_{s=1}^{s=N_{sd}} B_{s,c}^t f_{s,c} - \sum_{s=1}^{s=N_x} B_{s,c}^t K_{s,rc}^t K_{s,rr}^{-1} f_{s,r}. \quad (11)$$

$B_{s,c}$ and $B_{s,r}$ are boolean matrices used to extract the primal c and rest components r of subdomain s .

A. Regularization

Because of the subdivision into subdomains, the previous techniques to remove null pivots are difficult to apply. For this work, we propose the following regularization technique, which is more adapted to the domain decomposition approach. Instead of solving (2), the following problem is considered [6]

$$\text{curl}(\mu^{-1} \text{curl} \mathbf{a}) + \sigma \mathbf{a} = \mathbf{j}. \quad (12)$$

The regularization coefficient σ has positive constant value in each subdomain Ω_i . Different coefficients have been tested. The term $\sigma \mathbf{a}$ can be added at the integration step when building the matrix [7] or algebraically after. The regularization should enable us to have a non-singular problem while maintaining good results.

B. Application

Regularization has been implemented in Altair FluxTM software [8]. Figure 1 illustrates the test case and represents a simplified 3D magnetic contactor decomposed into 4 subdomains. The cube in the middle is a linear magnet. The other two regions represent ferromagnetic material. All three regions are surrounded by an air region.

Different values of the regularization coefficient have been tested for various coarse problem sizes. We studied the matrix

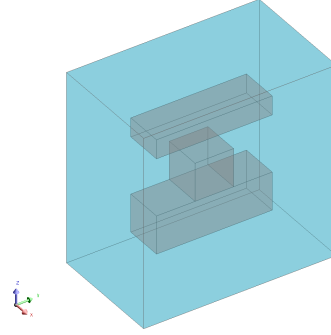


Fig. 1. Test case of 3D magnetostatic simulations with edge elements

spectrum and the conditioning for the full problem, subdomains, interface and coarse problems.

The assembled matrix for the full problem is ill-conditioned because many eigenvalues are close to zero but not exactly zero. Our study consisted of analysing the interface problem matrix conditioning. Different values of the regularization coefficient, ranging from 1×10^{-7} to 1×10^7 were used. The higher the regularization coefficient, the better the results (smaller condition number). Examples have shown that there is a maximum value that can be taken, and exceeding this value results in an error compared to standard solving. The size of the coarse problem is also important. Domain decomposition method implies to have a coarse problem as small as possible. At the same time, our analysis shows that the bigger the coarse problem, the better the conditioning.

IV. CONCLUSION

The decomposed aspect of domain decomposition can result in difficulties in magnetostatic simulations using edge elements. In this work, we explored a technique to address this issue, which is simple to implement and yields good results. Nevertheless, it also introduces input variables (coarse problem size and a regularization coefficient) that need to be chosen carefully. We applied it to a first test case and foresee applying it to more difficult cases.

REFERENCES

- [1] G. Meunier, ed., "The finite element method for electromagnetic modeling," John Wiley & Sons, 2010, pp. 69-83.
- [2] P. Ferrouillat, "Développement de formulations éléments finis 3D en potentiel vecteur magnétique: application aux machines asynchrones en mouvement," PhD thesis. Université Grenoble Alpes, 2015.
- [3] N. Spillane, "Méthodes de décomposition de domaine robustes pour les problèmes symétriques définis positifs," PhD thesis. Paris 6, 2014.
- [4] C. Farhat, and F. X. Roux, "A method of finite element tearing and interconnecting and its parallel solution algorithm," International journal for numerical methods in engineering 32 (1991), pp. 1205-1227.
- [5] C. Farhat, M. Lesoinne, P. LeTallec, K. Pierson and D. Rixen, "FETI-DP: a dual-primal unified FETI method—part I: A faster alternative to the two-level FETI method." International journal for numerical methods in engineering 50.7 (2001): 1523-1544.
- [6] A. Toselli, "Dual-primal FETI algorithms for edge finite-element approximations in 3D." IMA journal of numerical analysis 526.1 (2006): 96-130.
- [7] S. Reitzinger, and J. Schöberl, "An algebraic multigrid method for finite element discretizations with edge elements," Numerical linear algebra with applications, 2002, vol. 9, no 3, p. 223-238.
- [8] Altair Engineering, "Altair FluxTM," <https://www.altair.com/flux>.