Struct Multidisc Optim (2008) 36:329–345
DOI 10.1007/s00158-007-0190-7

RESEARCH PAPER

Large-scale parallel topology optimization using a dual-primal substructuring solver

**Anton Evgrafov · Cory J. Rupp ·
Kurt Maute · Martin L. Dunn**

Received: 30 January 2007 / Revised: 4 September 2007 / Accepted: 12 September 2007 / Published online: 19 October 2007

Abstract Parallel computing is an integral part of many scientific disciplines. In this paper, we discuss issues and difficulties arising when a state-of-the-art parallel linear solver is applied to topology optimization problems. Within the topology optimization framework, we cannot readjust domain decomposition to align with material decomposition, which leads to the deterioration of performance of the substructuring solver. We illustrate the difficulties with detailed condition number estimates and numerical studies. We also report the practical performances of finite element tearing and interconnection/dual–primal solver for topology optimization problems and our attempts to improve it by applying additional scaling and/or preconditioning strategies. The performance of the method is finally illustrated with large-scale topology optimization problems coming from different optimal design fields: compliance minimization, design of compliant mechanisms, and design of elastic surface wave-guides.

Keywords Topology optimization · Parallel computing · Scalability · Domain decomposition · Iterative solvers

1 Introduction

Topology optimization, or control in coefficients, has become an increasingly popular methodology for systematically computing novel designs in many engineering disciplines owing to the flexible parametrization of the design space that allows optimization algorithms to efficiently explore it. Other advantages of using topology optimization include the ease of integration with existing computational codes in a variety of application areas and the simplicity and efficiency of sensitivity analyses. Since the pioneering paper by Bendsøe and Kikuchi (1988), this computational approach has seen tremendous development in academia and even some adoption by industry. The number of research papers on this topic is abundant and has been reviewed in a recent monograph (Bendsøe and Sigmund 2003). There are different schools of thought when it comes to handling computational complexity associated with solving the partial differential equations (PDEs) governing an engineering system being optimized, many of which

The authors acknowledge the support of the Air Force Office of Scientific Research (AFOSR) under grant FA9550-05-1-0046. The computational facility was obtained under the grant AFOSR-DURIP FA9550-05-1-0291.

A. Evgrafov (✉) · K. Maute
Center for Aerospace Structures,
Department of Aerospace Engineering Sciences,
University of Colorado,
Boulder, CO 80309–0429, USA
e-mail: antone@colorado.edu

K. Maute
e-mail: maute@colorado.edu

C. J. Rupp · M. L. Dunn
Department of Mechanical Engineering,
University of Colorado, Boulder,
CO 80309–0427, USA

C. J. Rupp
e-mail: Cory.Rupp@colorado.edu

M. L. Dunn
e-mail: Martin.Dunn@colorado.edu

concentrate on constructing computational meshes in an adaptive manner. However, some application areas require the use of uniform high-fidelity meshes to capture the physics of the problem, including high-frequency wave propagation problems, viscous flows at high Reynolds numbers in complicated geometries, and computational lattice-based methods, such as the lattice Boltzmann method. Time-wise, repeatedly solving PDEs on such high-resolution computational grids makes parallel computing a must and an integral part of any optimal design procedure for such applications.

No one would argue that the computational power has vastly increased since the publication of the first paper on topology optimization (Bendsøe and Kikuchi 1988) or since the publication of the popular 99 line Matlab educational code for topology optimization (Sigmund 2001); see Figs. 1a,b. Computing power has recently reached previously unimaginable number of $360 \cdot 10^{12}$ floating point operations per second (360 Tera-FLOP), and Peta-FLOP (10^{15} FLOP) computers are expected by the end of 2010. Furthermore, the cost of computer hardware has steadily decreased, and parallel computing systems have become widely available.

However, analysis of the publications in the area of topology optimization shows that the community does not yet fully utilize the available parallel computing hardware, as opposed to the situation in, for example, the computational fluid dynamics or numerical linear algebra communities. Furthermore, sizes of optimal design problems solved using topology optimization techniques and parallel computing reported in the literature (Borrvall and Petersson 2001; Kim et al. 2004; Mahdavi et al. 2006; Vemaganti and Lawrence 2005) become smaller and smaller in the recent years, while the number of processors used for parallel topology optimization remains essentially constant (see Table 1). The issues of efficient parallel computing for topology optimization in fluid mechanics were raised and to some extent addressed by Evgrafov et al. (2006) by switching from the finite element formulations to efficiently parallelizable finite difference-type methods. In this paper we will be concerned with more classical linear elasticity problems, both in statics and forced vibrations (quasi-statics), for which the finite element method (FEM) is the method of choice when it comes to discretizing the governing PDEs. Our goal is to analyze the performance of parallel solvers for design optimization purposes and to identify difficulties one faces when trying to solve large-scale topology optimization problems.

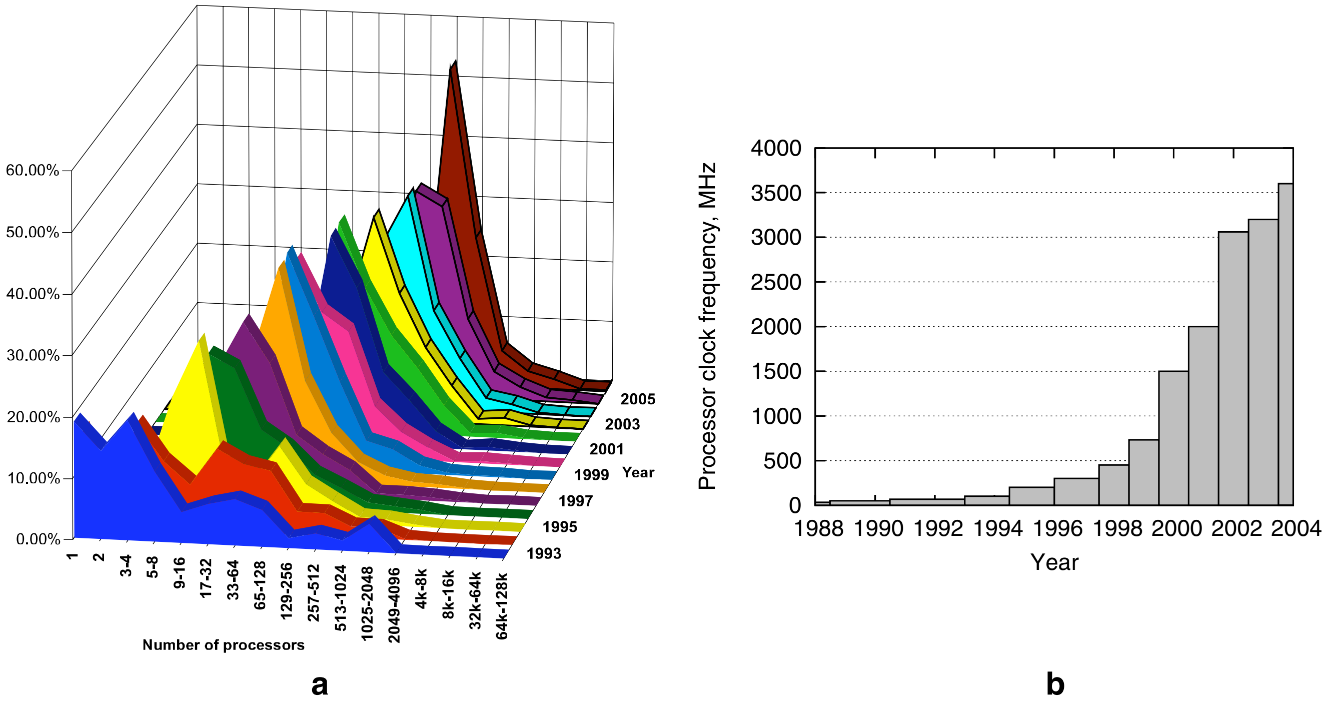

Fig. 1 **a** Distribution of the number of processors in world's top 500 supercomputers over the years, see http://www.top500.org/stats. **b** Top clock frequency of Intel processors over the past years, see http://www.intel.com/pressroom/kits/quickrefyr.htm. We emphasize that the diagram is displayed for the purpose of presenting the tendency only and does not take into account factors making the processors even more powerful such as the increase of cache sizes, bus sizes, queuing and pipelining strategies, and so on. Processors from other manufacturers also demonstrate steady growth in performance over the years

Table 1 Representative sizes of some topology optimization problems solved in recent years

Year	Reference	Max #DOFs	Max #proc	Spatial dim
2001	Sigmund (2001)	$2.4 \cdot 10^3$	1	2D
2001	Borrvall and Petersson (2001)	$6.63 \cdot 10^5$	32	3D
2004	Kim et al. (2004)	$4.07 \cdot 10^6$	16	3D
2005	Vemaganti and Lawrence (2005)	$1.15 \cdot 10^5$	32	2D
2006	Mahdavi et al. (2006)	$2.56 \cdot 10^4$	32	2D
2006	Wang et al. (2007)	$1.01 \cdot 10^6$	1	3D

For example, topology optimization is often employed at preliminary design stages (Duysinx and Bruyneel 2002), whence there is no need to use high-resolution finite element models. However, certain physical phenomena require high fidelity simulations to capture the physics of the problem. In fact, planar optimal designs for wave propagation problems do not require any postprocessing and can be manufactured "as is" (Bendsøe 2006). We emphasize that the word "planar" here refers to the design only, and in particular, the case of surface waves requires full three-dimensional (3D) high fidelity finite element modeling (Rupp et al. 2006).

Furthermore, *numerical scalability* has not been properly addressed in parallel topology optimization papers. Generally, the computational science community distinguishes between *parallel* and *numerical* scalabilities. Parallel scalability characterizes the ability of an algorithm, its parallel implementation, and a given parallel computer to demonstrate a speedup that increases with the number of processors *for a fixed problem size*. This is the kind of scalability that is usually investigated in papers on parallel topology optimization, but only on a limited number of processors, which does not allow one to reveal the true amount of overhead related to inter-processor communication. However, arguably, the objective of using parallel computers for topology optimization purposes is to make possible solution of large-scale 3D problems rather than to solve fixed size problems faster, and here, the notion of *numerical scalability* comes in handy. A solution method is said to be numerically scalable if its computational complexity grows asymptotically linearly with the size of the problem. For example, it is relatively simple to implement a finite element assembly routine and a conjugate gradient (CG) iterative solver on a distributed memory parallel computer to achieve parallel scalability. However, if we start to increase the finite element mesh resolution and hence the problem size, more and more CG iterations will be required to achieve a prescribed and fixed accuracy. Thus, while we achieve numerical scalability *per CG step*, the whole solution procedure will not be numerically scalable. In particular, the domain decomposition algorithms cannot be numerically scalable, i.e., have a rate of convergence that is independent of the number of subregions, unless a coarse space component is included (Klawonn et al. 2002). Therefore, extrapolating the number of iterations required to solve the finite element system using a CG algorithm as reported in Mahdavi et al. (2006), Vemaganti and Lawrence (2005), Wang et al. (2007) onto a larger number of parallel processors might be meaningless. In some works (Mahdavi et al. 2006), the computational performance is illustrated only on 2D problems; however, completely different preconditioning strategies must be used to keep iterative solvers numerically scalable in 2D and 3D, again, limiting practical applicability of such results.

At last, we would like to demonstrate that simply having a state-of-the-art, both parallel and numerically scalable linear solver is not sufficient to efficiently solve topology optimization problems on a parallel computer. As we will show, this is owing to the unusual scaling of resulting linear systems, especially those resulting from solid–void optimal design problems in linearized elasticity with simple isotropic material with penalization (SIMP; Bendsøe and Sigmund 2003; Rozvany et al. 1992). The reason for such a behavior is that the domain decompositions used within topology optimization cannot be aligned with material distributions and that the latter are not clearly defined and change after every optimization step.

For the purpose of this paper, we have chosen a state-of-the-art linear finite element tearing and interconnection/dual primal solver (FETI-DP; Farhat et al. 2001) specifically designed to solve large-scale systems arising from 2D/3D high fidelity finite element models on modern distributed memory parallel computers. The FETI-DP algorithm belongs to the class of domain decomposition methods, the best known and perhaps the most promising parallel linear solvers revolving around the concept of graph partitioning (Saad 2003). Variants of FETI methods are widely adopted by the scientific computing community (see, e.g., Proceedings of the International Conferences on Domain Decomposition Methods) and the industry (a variant of FETI has been implemented as a part of Parallel Performance for ANSYS products (O'Neal and Murgie 2001)). Their success can be attributed, similarly to the popularity of the interior point methods in optimization, both to the excellent performance in many practical situations and to the existence of the theoretical upper bound asserting the numerical scalability of the methods, which holds at least in the case of various elliptic PDE problems (Klawonn et al. 2002; Mandel and Tezaur 2001). However, this bound is not very practical for topology optimization applications, as we will discuss in Section 4, and the difficulty stems from the heterogeneity of PDE coefficients within the subdomains, resulting from the control in coefficients structure of topology optimization problems and changing with every optimization iteration. Furthermore, the practical performance of the method also deteriorates, showing that the upper bound on the number of iterations is not just a pessimistic estimate. Therefore, one has either to realign subdomain boundaries with material boundaries after each optimization iteration, which is not practical and defeats the whole purpose of topology optimization, or to modify the preconditioning strategy taking into account the heterogeneity within the subdomains. We find that variations of the Jacobi-type preconditioners that improve the condition number of the systems of linear equations resulting from the FEM discretization of the elasticity equations (Mahdavi et al. 2006; Vemaganti and Lawrence 2005; Wang et al. 2007) only marginally improve the performance of FETI-DP in the topology optimization context. We study the performance of FETI-DP for topology optimization problems with 2D and 3D examples and report our computational experiences.

All numerical tests presented in this paper are performed on a 64-bit Linux cluster composed of 32 computational nodes, interconnected with an InfiniBand-based network. Each computational node has four AMD Opteron 846 HE processors (featuring a 2-GHz clock speed and 1 MB of L2 cache) and 8 GB of memory.

The organization of this paper is as follows. In Section 2, we touch briefly on the formulation of the topology optimization problems. In Section 3, we provide a minimalistic description of the FETI-DP algorithm to keep this paper self-contained. In Section 4, we discuss a refined version of the numerical scalability estimate for FETI-DP, which explicitly accounts for the heterogeneity of the subdomains. In Section 5, additional Jacobi-type preconditioning strategies and their effect on the performance of FETI-DP are discussed. Section 6 is dedicated to solving large-scale topology optimization problems to illustrate the capabilities of the presented computational methodology. Both definite (for which the theoretical scalability is expected) and indefinite systems are solved. We finish the paper with a brief discussion and possible further research directions.

2 Topology optimization problems

In many practical applications (linearized elasticity, linear wave propagation, and Stokes flows), a topology optimization problem may be stated as follows. Let $\Omega \in \mathbb{R}^d$ be a bounded regular domain. In this domain, we consider the following variational problem:

$$\begin{aligned} &\underset{(\rho,\mathbf{u})\in L^\infty(\Omega)\times H^1(\Omega)}{\text{minimize}} \quad \mathscr{F}(\rho,\mathbf{u}),\\ &\text{subject to } \langle \mathscr{A}(\rho)\mathbf{u}, v\rangle = \langle f, v\rangle, \quad \forall v \in H^1_0(\Omega)\\ &\qquad (\rho,\mathbf{u}) \in \mathscr{Z}. \end{aligned} \tag{1}$$

In this study, the variable $\rho \in L^\infty(\Omega)$ plays a rôle of the design being optimized, and it describes a distribution of some material within Ω; usually we can scale ρ so that $0 \le \rho \le 1$, a.e. The variable $\mathbf{u} \in H^1(\Omega)$, on the other hand, describes a state of the mechanical system under the consideration, corresponding to a given design ρ. The bilinear form $\langle \mathscr{A}(\rho)\mathbf{u}, v\rangle$ is usually regular enough for the design-to-state mapping $\rho \rightrightarrows \{\mathbf{u} \in H^1(\Omega) \mid \langle \mathscr{A}(\rho)\mathbf{u}, v\rangle = \langle f, v\rangle,\ \forall\ v \in H^1_0(\Omega)\}$ to be single-valued for every feasible design ρ. The set of *simple* constraints $\mathscr{Z} \subset L^\infty(\Omega) \times H^1(\Omega)$ is assumed to be sufficiently restrictive (e.g., in linear elasticity problems, it is often assumed that $\|\rho\|_{BV(\Omega)} \le C$, known as the perimeter constraint; Petersson 1999) to guarantee the closedness of the design-to-state mapping in a suitable topology. Similarly, the performance functional $\mathscr{F} : L^\infty(\Omega) \times H^1(\Omega) \to \mathbb{R}$ is assumed to be coercive and lower semi-continuous in a suitable topology so that Weierstrass' theorem (see, e.g., Rockafellar and Wets 1998) is applicable to problem (1).

For computational purposes, the problem (1) is discretized using a FEM, resulting in the following nonlinear programming problem:

$$\begin{aligned} &\underset{(\rho,\mathbf{u})\in \mathbb{R}^m\times\mathbb{R}^n}{\text{minimize}} \ \mathscr{F}(\rho,\mathbf{u}),\\ &\text{subject to } \mathsf{K}(\rho)\mathbf{u} = f,\\ &\qquad (\rho,\mathbf{u}) \in \mathscr{Z}, \end{aligned} \tag{2}$$

where the matrix $\mathsf{K} : \mathbb{R}^m \to \mathbb{R}^{n\times n}$ is assumed to be nonsingular and to enjoy the following property:

$$\mathsf{K}(\rho) = \mathsf{K}_0 + \sum_{i=1}^{m} \mathsf{K}_i(\rho_i). \tag{3}$$

For example, in the case of linearized elasticity with SIMP (Bendsøe and Sigmund 2003), the equation (3) takes an especially simple form:

$$\mathsf{K}(\rho) = \mathsf{K}_0 + \sum_{i=1}^{m} \rho_i^{\gamma} \mathsf{K}_i, \tag{4}$$

where in typical applications, we have $\mathsf{K}_i \succeq 0$, $10^{-3} \lessapprox \rho_i \leq 1$ and $\gamma \approx 3.0$.

Most often, the problem (2) is transformed into a so-called nested formulation:

$$\begin{aligned} &\underset{\rho\in\mathbb{R}^m}{\text{minimize}}\, \mathscr{F}(\rho, \mathsf{K}^{-1}(\rho)\mathbf{f}), \\ &\text{subject to } (\rho, \mathsf{K}^{-1}(\rho)\mathbf{f}) \in \mathscr{X}, \end{aligned} \tag{5}$$

which requires solving a linear system with the matrix $\mathsf{K}(\rho)$ to evaluate the values of the objective function and the constraints, as well as their gradients (see Bendsøe and Sigmund 2003). We employ FETI-DP for this purpose.

3 FETI-DP method

In the present and in the following sections, we discuss the formulation of and the condition number estimate for FETI-DP algorithm for elliptic problems. We do this in two spatial dimensions following Mandel and Tezaur (2001) to simplify the presentation, although the generalization for 3D problems is possible as shown in Klawonn et al. (2002), as well as indefinite problems as addressed in Farhat et al. (2005).

Let Ω be a regular domain in $\mathbb{R}^2$, where an elliptic definite system of PDEs is considered. We decompose the domain into N_s nonoverlapping regular subdomains $\Omega^1, \Omega^2, \ldots, \Omega^{N_s}$, also known as substructures, where each subdomain is a union of a number of shape regular finite elements of size h. We let $H_s = \operatorname{diam}(\Omega^s)$, $s = 1, \ldots, N_s$, and assume that $H_1 \approx H_2 \approx \cdots \approx H_{N_s} \approx H$. Let $\mathbf{u}^s$ be a vector of degrees of freedom (DOFs) for the subdomain Ω^s corresponding to a chosen finite element discretization. Similarly, let K^s and f^s be the local stiffness matrix and the load vector associated with the subdomain Ω^s. The edges of the subdomains will be denoted by $\Gamma^{st} := \partial\Omega^s \cap \partial\Omega^t$, and corners are the endpoints of edges.

The subdomain vectors are then partitioned as

$$\mathbf{u}^s = \left[(\mathbf{u}_i^s)^T, (\mathbf{u}_r^s)^T, (\mathbf{u}_c^s)^T\right]^T,$$

where $\mathbf{u}_i^s$ are the values of the DOFs in the subdomain interior, $\mathbf{u}_c^s$ the values of the DOFs at the corners of the subdomain, and $\mathbf{u}_r^s$ are the remaining values of the DOFS, i.e., those located on the edges of the subdomains between the corners. The subdomain matrices are partitioned accordingly,

$$\mathsf{K}^s = \begin{pmatrix} \mathsf{K}_{ii}^s & \mathsf{K}_{ir}^s & \mathsf{K}_{ic}^s \\ \mathsf{K}_{ri}^s & \mathsf{K}_{rr}^s & \mathsf{K}_{rc}^s \\ \mathsf{K}_{ci}^s & \mathsf{K}_{cr}^s & \mathsf{K}_{cc}^s \end{pmatrix}.$$

We also define the block-vectors and matrices:

$$\mathbf{u} = \begin{pmatrix} \mathbf{u}^1 \\ \vdots \\ \mathbf{u}^{N_s} \end{pmatrix}, \quad \mathsf{K} = \begin{pmatrix} \mathsf{K}^1 & \ldots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \ldots & \mathsf{K}^{N_s} \end{pmatrix}, \quad \text{and} \quad f = \begin{pmatrix} f^1 \\ \vdots \\ f^{N_s} \end{pmatrix}.$$

Similarly,

$$\mathbf{u}_i = \begin{pmatrix} \mathbf{u}_i^1 \\ \vdots \\ \mathbf{u}_i^{N_s} \end{pmatrix}, \quad \mathbf{u}_c = \begin{pmatrix} \mathbf{u}_c^1 \\ \vdots \\ \mathbf{u}_c^{N_s} \end{pmatrix}, \quad \text{and} \quad \mathbf{u}_r = \begin{pmatrix} \mathbf{u}_r^1 \\ \vdots \\ \mathbf{u}_r^{N_s} \end{pmatrix}.$$

The DOFs from both sides of each edge Γ^{st} should coincide,

$$\mathbf{u}_r^s|_{\Gamma^{st}} - \mathbf{u}_r^t|_{\Gamma^{st}} = \mathbf{0}. \tag{6}$$

In (6), each pair of subdomains $\{s, t\}$ is taken only once, with the order (s, t) chosen arbitrarily. We write the constraint (6) as

$$\left(\mathsf{B}_r^1 \ldots \mathsf{B}_r^{N_s}\right) \mathbf{u}_r = \mathsf{B}_r \mathbf{u}_r = \mathbf{0}. \tag{7}$$

Let L_c be a matrix with $\{0, 1\}$ entries implementing the global-to-local map at the subdomain corners. That is the equation

$$\mathbf{u}_c = \mathsf{L}_c \mathbf{u}_c^g, \qquad \mathsf{L}_c = \begin{pmatrix} \mathsf{L}_c^1 \\ \vdots \\ \mathsf{L}_c^{N_s} \end{pmatrix},$$

determines the common values of the DOFs at the subdomain corners from a global vector $\mathbf{u}_c^g$.

In this notation, we are interested in solving the problem:

$$\begin{aligned} &\underset{\mathbf{u}}{\text{minimize}} \; \frac{1}{2}\mathbf{u}^T \mathsf{K} \mathbf{u} - f^T \mathbf{u}, \\ &\text{subject to } \; \mathsf{B}_r \mathbf{u}_r = \mathbf{0}, \text{ and } \mathbf{u}_c = \mathsf{L}_c \mathbf{u}_c^g, \text{ for some } \mathbf{u}_c^g. \end{aligned}$$

Equivalently, we are interested in finding a stationary point for the Lagrangian

$$\mathscr{L}(\mathbf{u}_i, \mathbf{u}_r, \mathbf{u}_c^g, \lambda) = \frac{1}{2} v^T \mathsf{K} v - f^T v + \lambda^T \mathsf{B}_r \mathbf{u}_r, \tag{8}$$

where

$$v = \begin{pmatrix} v^1 \\ \vdots \\ v^{N_s} \end{pmatrix}, \qquad \text{and} \qquad v^s = \begin{pmatrix} \mathbf{u}_i^s \\ \mathbf{u}_r^s \\ \mathsf{L}_c^s \mathbf{u}_c^g \end{pmatrix}.$$

Eliminating $\mathbf{u}_i^s$, $\mathbf{u}_r^s$, $\mathbf{u}_c^g$ from the stationarity conditions for (8), we obtain a dual system of the type:

$$F\lambda = \mathbf{g}. \tag{9}$$

For example, knowing $\mathbf{u}_c^g$ and λ, we can solve the following subdomain problem in parallel to obtain $\mathbf{u}_i$, $\mathbf{u}_r$:

$$\begin{pmatrix} K_{ii}^s & K_{ir}^s \\ K_{ri}^s & K_{rr}^s \end{pmatrix} \begin{pmatrix} \mathbf{u}_i^s \\ \mathbf{u}_r^s \end{pmatrix} = \begin{pmatrix} f_i^s - K_{ic}^s L_c^s \mathbf{u}_c^g \\ f_r^s - B_r^{sT}\lambda - K_{rc}^s L_c^s \mathbf{u}_c^g \end{pmatrix},$$

and $\mathbf{u}_c^g$ is computed from λ as

$$K_{cc}^* \mathbf{u}_c^g = F_{I_{rc}}^T \lambda - f_c^*,$$

where the above problem is known as the *coarse* problem. This problem is *global* in the sense that it concerns the global vector $\mathbf{u}_c^g$, and to assemble K_{cc}^* and $F_{I_{rc}}^T\lambda - f_c^*$, we need contributions from *all* subdomains [see (10) below]. However, it is small enough (unless millions of subdomains are used) so that it may be solved using direct solvers on every node in a parallel computer. Let us define the matrices and vectors involved in the coarse problem:

$$F_{I_{rr}} = \sum_{s=1}^{N_s} (0\ B_r^s) \begin{pmatrix} K_{ii}^s & K_{ir}^s \\ K_{ri}^s & K_{rr}^s \end{pmatrix}^{-1} \begin{pmatrix} 0 \\ B_r^{sT} \end{pmatrix},$$

$$F_{I_{rc}} = \sum_{s=1}^{N_s} (0\ B_r^s) \begin{pmatrix} K_{ii}^s & K_{ir}^s \\ K_{ri}^s & K_{rr}^s \end{pmatrix}^{-1} \begin{pmatrix} K_{ic}^s \\ K_{rc}^s \end{pmatrix} L_c^s,$$

$$K_{cc}^* = K_{cc} - \sum_{s=1}^{N_s} L_c^{sT} \left(K_{ic}^{s\,T}\ K_{rc}^{s\,T}\right) \begin{pmatrix} K_{ii}^s & K_{ir}^s \\ K_{ri}^s & K_{rr}^s \end{pmatrix}^{-1} \begin{pmatrix} K_{ic}^s \\ K_{rc}^s \end{pmatrix} L_c^s,$$

$$\mathbf{d}_r = \sum_{s=1}^{N_s} (0\ B_r^s) \begin{pmatrix} K_{ii}^s & K_{ir}^s \\ K_{ri}^s & K_{rr}^s \end{pmatrix}^{-1} \begin{pmatrix} \mathbf{f}_i^s \\ \mathbf{f}_r^s \end{pmatrix},$$

$$\mathbf{f}_c^* = \mathbf{f}_c - \sum_{s=1}^{N_s} L_c^{sT} \left(K_{ic}^{s\,T}\ K_{rc}^{s\,T}\right) \begin{pmatrix} K_{ii}^s & K_{ir}^s \\ K_{ri}^s & K_{rr}^s \end{pmatrix}^{-1} \begin{pmatrix} \mathbf{f}_i^s \\ \mathbf{f}_r^s \end{pmatrix}. \tag{10}$$

Notice that all calculations involve inverses of the subdomain matrices K^s, $s = 1, \ldots, N_s$; if we keep the size of the subdomains small, we may apply direct solvers in parallel to compute a factorization of these matrices and efficiently solve the related systems. The fact that many computations are completely decoupled to the subdomain level contributes to the parallel scalability of FETI-DP. In the notations of (10), the matrix F and the vector g defining the dual problem (9) may be written as:

$$F = F_{I_{rr}} + F_{I_{rc}} (K_{cc}^*)^{-1} F_{I_{rc}}^T,$$

$$\mathbf{g} = \mathbf{d}_r - F_{I_{rc}} (K_{cc}^*)^{-1} \mathbf{f}_c^*.$$

The dual, or *interface*, problem is solved using a CG (for elliptic definite problems) or other iterative solver, which means that the matrix F is never explicitly computed. What makes FETI-DP numerically scalable is the availability of an efficient preconditioner for (9):

$$M = B_r S_{rr} B_r^T, \tag{11}$$

where

$$S_{rr} = \begin{pmatrix} S_{rr}^1 & \ldots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \ldots & S_{rr}^{N_s} \end{pmatrix},$$

and

$$S_{rr}^s = K_{rr}^s - K_{ri}^s (K_{ii}^s)^{-1} K_{ir}^s.$$

Thus, in FETI-DP, we solve the system (9) with a preconditioned iterative solver (CG for elasticity problems, generalized minimal residual for wave propagation problems, etc.) with a preconditioner (11). The algorithm terminates successfully when the norms of the dual residual

$$\mathbf{r}_d = \|F\lambda - \mathbf{g}\|,$$

and the primal residual

$$\mathbf{r}_p = \|K\mathbf{u} - \mathbf{f}\|,$$

are within the prescribed tolerance. For details and the numerical implementation, the interested reader is referred to Farhat et al. (2001, 2005), Klawonn et al. (2001), Mandel and Tezaur (2002).

4 Conditional number estimates

FETI-DP is just one of the many possible methods based on general ideas of domain decomposition and Schur complements. What makes it attractive is the existence of theoretical estimates on the condition number of the preconditioned interface problem (9) thereby asserting its numerical scalability. In particular,Mandel and Tezaur (2001, Theorems 4.4, 4.5, 5.2) have established the following result:

Theorem 1 *For the second order elliptic positive definite problem*

$$\begin{aligned} -\sum_{i,j=1}^{2} \frac{\partial}{\partial x_i}\left(\rho(\mathbf{x})\frac{\partial u(\mathbf{x})}{\partial x_j}\right) &= g \qquad \textit{in } \Omega, \\ u &= 0 \qquad \textit{on } \partial\Omega, \end{aligned} \tag{12}$$

where $\rho : \Omega \to \mathbb{R}$ is a measurable function such that $0 < \rho_0 \leq \rho \leq \rho_1 < +\infty$, a.e. in Ω, it holds that

$$\mathrm{cond}(\mathsf{MF}) \leq C n_e [1 + \log(H/h)]^2, \tag{13}$$

where n_e is the maximal number of edges on any subdomain, and C is independent from h, H (but, generally, depends on ρ).

One of the basic moments in the proof is the equivalence of the Sobolev seminorms $|I_{P1}\mathbf{u}^s|_{\frac{1}{2},2,\partial\Omega^s}$ with the seminorm $|\mathbf{u}^s|_{\mathsf{S}^s}$, see (Bramble et al. 1986, Lemma 3.1), where $I_{P1}\mathbf{u}^s$ is the function from the restriction of the finite element space $V_h^{P1}(\Omega)$ onto Ω^s, defined by the vector of DOFs $\mathbf{u}^s$,

$$|u|^2_{\frac{1}{2},2,\Gamma} = \iint_\Gamma \frac{|u(x) - u(y)|^2}{|x - y|^2}\, dx\, dy,$$

$$|\mathbf{u}|_{\mathsf{A}} = \| A^{1/2}\mathbf{u} \|, \quad \text{for a positive semidefinite matrix } \mathsf{A}, \text{ and}$$

$$(\mathbf{u}^s)^T \mathsf{S}^s \mathbf{u}^s = \min\{ (\mathbf{v}^s)^T \mathsf{K}^s \mathbf{v}^s \mid \mathbf{v}^s_c = \mathbf{u}^s_c \text{ and } \mathbf{v}^s_r = \mathbf{u}^s_r \}.$$

From the proof of (Theorem 5.2 Mandel and Tezaur 2001) and (Lemma 3.1 Bramble et al. 1986), it follows that the constant c appearing in the Theorem 1 actually depends on the constants estimating the equivalence of the seminorms, which are:

$$c_1 [\operatorname*{ess\,sup}_{\mathbf{x}\in\Omega_s} \rho(x)]^{-1} |\mathbf{u}^s|^2_{\mathsf{S}^s} \\ \leq |I_{P1}\mathbf{u}^s|^2_{\frac{1}{2},2,\partial\Omega^s} \leq c_2 [\operatorname*{ess\,inf}_{\mathbf{x}\in\Omega_s} \rho(x)]^{-1} |\mathbf{u}^s|^2_{\mathsf{S}^s},$$

where c_1, c_2 are positive constant independent of h, H, ρ. Therefore, it holds that

$$\mathrm{cond}(\mathsf{MF}) \leq \widetilde{c} n_e [1 + \log(H/h)]^2 \max_{1 \leq s \leq N_s} \frac{\mathrm{ess\,sup}_{\mathbf{x}\in\Omega_s} \rho(x)}{\mathrm{ess\,inf}_{\mathbf{x}\in\Omega_s} \rho(\mathbf{x})}, \tag{14}$$

where $\widetilde{c}$ is a generic constant independent of h, H, ρ. Similar reasoning can be applied to other elliptic definite problems appearing in linearized elasticity (Klawonn et al. 2002; Mandel and Tezaur 2001); computational experience shows that the method is comparably scalable for forced vibrations problems (Farhat et al. 2005).

Let us now discuss the implications of the refined estimate (14) in the context of topology optimization (2) with SIMP (4). During the progress of the optimization algorithm, the last factor in the estimate (14) changes from 1.0 for the "standard" homogeneous initial design to $(\overline{\rho}/\underline{\rho})^\gamma$, where $\overline{\rho}$ and $\underline{\rho}$ are the densities corresponding to the stiff and the soft materials, and γ is a SIMP factor. In typical applications, we set $\overline{\rho} = 1.0$, $\underline{\rho} \approx 10^{-3}$, and $\gamma \approx 3.0$; therefore, if the upper estimate (14) is not too pessimistic, we should expect significant degeneration of the rate of convergence of FETI-DP as the optimization algorithm progresses (see Fig. 3b). Perhaps even more discouraging, while the estimate (14) still formally asserts the linear complexity whence the numerical scalability of FETI-DP, the proportionality coefficient of $(\overline{\rho}/\underline{\rho})^\gamma \approx 10^9$ for typical structural topology optimization problems makes this estimate less than practical.

In the following sections, we report the practical performance of FETI-DP for topology optimization problems and our attempts to improve it by applying additional scaling and/or preconditioning strategies.

5 Additional scaling for FETI-DP

To remedy the uncontrollable growth of the condition number of the dual problem in FETI-DP (and hence the number of CG iterations needed to solve it) within the topology optimization context as given by the estimate (14), we study a few different strategies. We test symmetric Jacobi-type preconditioning, rescaling of the primal residual (which is equivalent to non-symmetric Jacobi-type preconditioning), and scaling of the augmentation matrix (see below). These modifications however do not improve the performance of FETI-DP enough to use it for topology optimization problems with stiff/void materials and SIMP, but we feel it is important to report our computational experience to the structural optimization community.

In addition to various scalings, we also implement a couple of convergence monitors, which are not specific to FETI-DP and are intended to improve the robustness of the whole optimization algorithm should the iterative solver fail to reduce the residual of the direct or adjoint systems to a prescribed accuracy at some optimization iteration. First of all, all iterative solvers are applied to the interface problem and, whence, are intended to make the dual residual as small as possible. Unfortunately, for poorly scaled problems such as the classical compliance minimization problems with SIMP and stiff/void materials, the primal residual may increase during the course of the iterations. As a result, should the algorithm be terminated prematurely, the primal solution corresponding to the last iteration may not be the best one. Thus, during the

FETI-DP iterations, we keep track of the best primal solution observed so far and, upon termination, return this solution. We also terminate the algorithm prematurely should the primal residual exceed the best observed primal residual by a prescribed factor (usually $\sim 10^2$).

5.1 Benchmarks

To illustrate the performance of FETI-DP for topology optimization applications, we use a "standard" benchmark, the so-called Messerschmitt–Bölkow–Blohm (MBB)-beam (Bendsøe and Sigmund 2003) and its 3D analogue. These examples are presented here for the reader's convenience.

In both 2D Fig. 2a and 3D Fig. 2d cases, we are interested in maximizing the stiffness of the structure or, equivalently, in minimizing the compliance of the solid measured as $f^T\mathbf{u}$, with f being the FEM-discretized vector of the external forces and $\mathbf{u}$ the nodal displacements, respectively. In both cases, we use filters to achieve mesh independency (Bourdin 2001), the maximum available volume of material is 30% of the control volume, SIMP exponent is $\gamma = 3.0$, and each design variable is bounded as $0.05 \leq \rho_i \leq 1$, unless specifically mentioned otherwise.

In the following subsections, we will refer to "optimal" domain decompositions, which will be understood as those keeping the subdomains geometrically regular and "minimizing" the length of the interface between the subdomains thus making the size of the dual problem smaller, see Fig. 2c. We use the method of moving asymptotes (MMA) (Svanberg 1987) as the optimization algorithm, the starting point being a homogeneous distribution of density throughout the control volume, in accordance with the standard practice in structural and multidisciplinary optimization. We note that the behavior of the underlying linear solver, which is of main interest to us, does not depend on the use of a particular optimization algorithm; however, separable

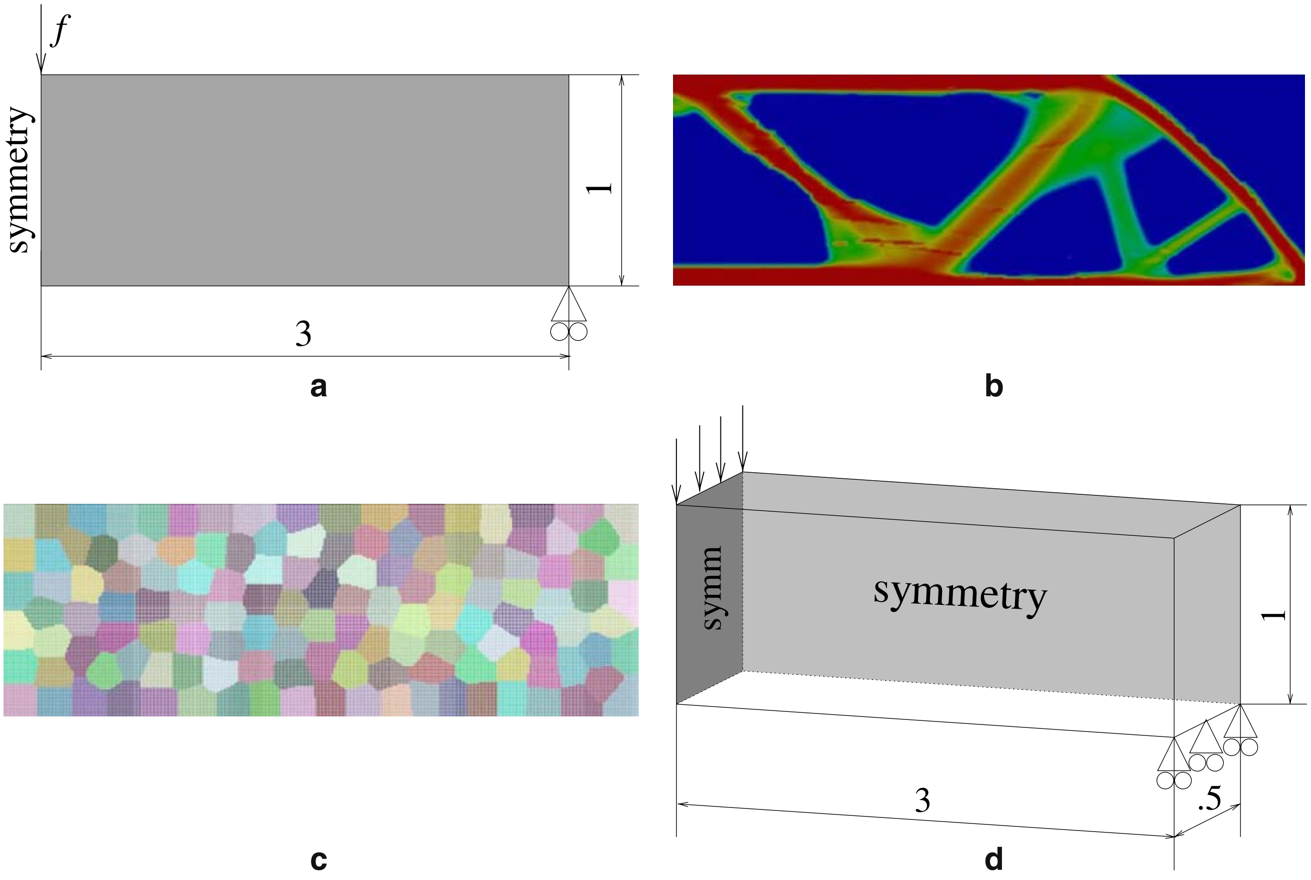

Fig. 2 Benchmarks for FETI-DP solver applied to topology optimization problems. **a** Setup for the 2D MBB-beam problem. **b** "Optimal" 2D MBB-beam; this solution has been obtained by running MMA with a direct solver and is intentionally not fully convergent for the purpose of studying the performance of FETI-DP on such material distributions. **c** "Optimal" 2D domain decomposition. **d** Setup for the 3D MBB-beam problem

convex approximation presents several advantages for large-scale optimization, such as its modest memory requirements (Svanberg 2002) and the ease of parallelization.

In 2D, we also study the performance of FETI-DP on "nearly optimal" designs, see Fig. 2b. Depending on the settings on the lower density limit $\underline{\rho}$ and filter parameters, one may obtain differing designs. We also study the behavior of FETI-DP on completely "black–white" (solid–void) designs, which we obtain from Fig. 2b via the simplest possible postprocessing procedure; that is, we let density variables whose values fall below a chosen threshold to be $\underline{\rho}$ and set the rest to be $\overline{\rho}$.

We set the precision for FETI-DP; that is, the stopping criteria for CG solver, to be 10^{-7} for the purposes of this paper. This implies that, if successfully stopped, FETI-DP exits with primal and dual solutions making the relative error in terms of primal and dual residuals, respectively, below 10^{-7}. What is the required precision for a given topology optimization problem is a question deserving discussion. First of all, having small primal and dual residuals does not guarantee that the relative error in calculating the displacements is equally small (i.e., $\|\mathbf{u}-\mathbf{u}^*\|/\|\mathbf{u}^*\| \leq 10^{-7}$, where $\mathbf{u}$ is the solution produced by FETI-DP and $\mathbf{u}^*$ is the exact solution to the system). This distinction is even more important for ill-scaled problems such as topology optimization problems with SIMP. Namely, we can say that we calculate $\mathsf{K}\mathbf{u}$ rather precisely; therefore, if one is interested in the compliance of the structure, $1/2\mathbf{u}^t\mathsf{K}\mathbf{u}$, or similar "energy-type" integral functionals, small error in the primal residual will result in a small error in the computed functional. This is the case for classical compliance minimization problems or, to lesser extent, in wave-guiding applications where the elastic energy integral is computed over certain parts of the domain only. On the other hand, if an optimization problem contains objectives or constraints that are *local*, such as the individual nodal displacements, or stresses, perhaps, even smaller residual norms will be required. We illustrate this difference in Section 6.

Apart from the difference between the local and the global constraints, the required precision in the optimal solution (measured in terms of the residual of KKT conditions, for example) will also determine the required accuracy of the linear solves.

5.2 Symmetric Jacobi-type preconditioning

Ill-conditioning appearing as a result of highly heterogeneous elastic properties is well understood. Computational experience suggests that simple Jacobi preconditioning is rather efficient for such systems (Mahdavi et al. 2006; Vemaganti and Lawrence 2005; Wang et al. 2007) [as opposed to the situation arising in nearly incompressible hydrodynamics (Lee et al. 2002)].

Thus, instead of using the original stiffness matrix in the problem (2), we solve the following prescaled equation:

$$\left[\mathsf{D}(\rho)\mathsf{K}(\rho)\mathsf{D}(\rho)\right]\left[\mathsf{D}^{-1}(\rho)\mathbf{u}\right] = \left[\mathsf{D}(\rho)f\right],$$

where

$$\mathsf{K}(\rho) = \begin{pmatrix} k_{11}(\rho) & \dots & k_{1n}(\rho) \\ \vdots & \ddots & \vdots \\ k_{n1}(\rho) & \dots & k_{nn}(\rho) \end{pmatrix}, \quad \text{and}$$

$$\mathsf{D}(\rho) = \begin{pmatrix} k_{11}^{-1/2}(\rho) & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & k_{nn}^{-1/2}(\rho) \end{pmatrix}. \tag{15}$$

One of the issues with the preconditioner $\mathsf{D}(\rho)$ is that it requires the knowledge of the coefficients $k_{ii}(\rho)$, $i = 1, \dots, n$, before the preconditioning of $\mathsf{K}(\rho)$ and f may be performed. Arguably, it would be much easier to precondition the matrix and the right-hand side of the system on an elemental level, i.e., before the system matrix K^s is assembled on each subdomain.

Fortunately, we know the structure of the matrix $\mathsf{K}(\rho)$ very well. For example, in the case of the linearized elasticity, we know that the "problematic" DOFs are actually those completely surrounded by the "soft" or "nearly void" elements. Therefore, we may redefine the diagonal elements of the preconditioner $\mathsf{D}(\rho)$ as $d_{ii}(\rho) = C\|\rho_i^{\text{adj}}\|^{-\gamma/2}$, where ρ_i^{adj} is a (short) vector containing densities of the finite elements adjacent to the node associated with ith degree of freedom, and γ is the SIMP penalization factor. Thus, at the assembly stage, we only have to communicate the necessary design variables ρ to all subdomains, but the integration and the scaling/preconditioning is performed only at the elemental level, whence completely in parallel. Furthermore, it is easy to verify that $d_{ii}(\rho) \sim k_{ii}^{-1/2}(\rho)$ as $k_{ii}(\rho) \approx 0$, which is the critical case for us.

The effect of the symmetric Jacobi preconditioning on the number of CG iterations is shown in Fig. 3. For this study, we use the 2D MBB-beam problem as discussed in Section 5.1. Thus, in this case, we deal with a material properties ratio for stiff and soft materials of "only" (in topology optimization terms) $8.0 \cdot 10^3$. Lower bounds on the material density result in much

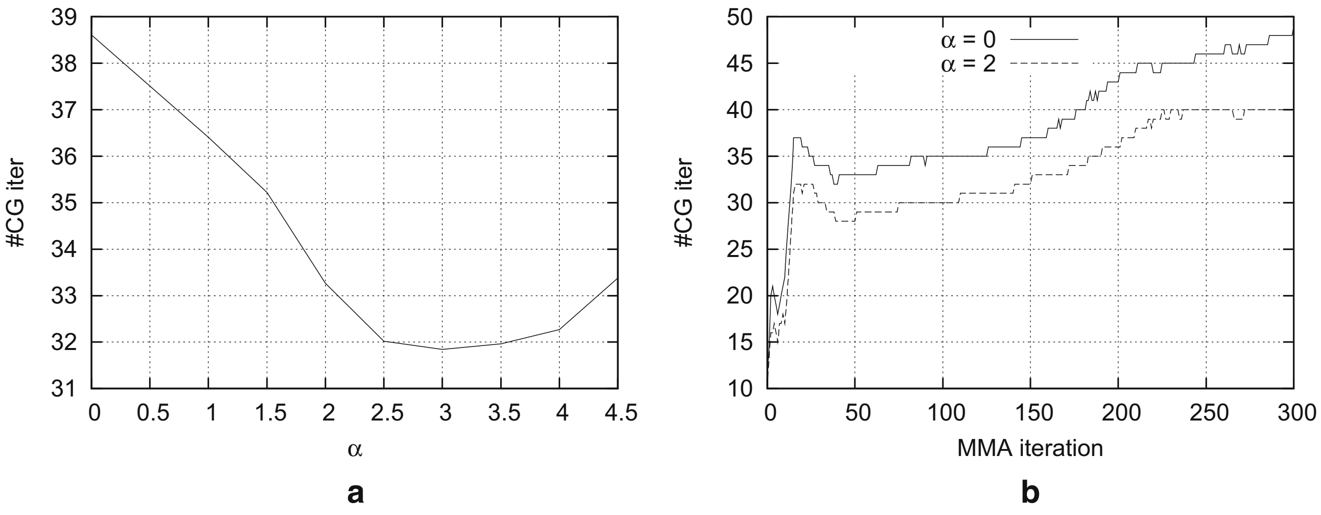

Fig. 3 Effect of the symmetric Jacobi-type preconditioning on the number of CG iterations in FETI-DP. **a** Average number of CG iterations over 300 MMA iterations; **b** increase in the number of CG iterations with increasing heterogeneity of the subdomains

worse performance (in many cases, the algorithm failed to reduce the residuals to a given tolerance). In Fig. 3a, we display an averaged number of CG iterations over 300 MMA (Svanberg 1987) iterations as a function of α defining the diagonal preconditioner via $d_{ii}(\rho) \sim \|\rho_i^{\text{adj}}\|^{-\alpha}$. One may see that (at least for this particular problem and given lower density limit) the minimum is not attained at the point $\alpha = \gamma/2 = 1.5$ as we expected from the asymptotics of the diagonal stiffness matrix elements $k_{ii}(\rho)$, but rather around $\alpha = 3.0$. This discrepancy is due to the fact that our lower limit $\underline{\rho}$ is relatively far from zero for the asymptotic analysis to hold. We have selected $\alpha = 2.0$ for our further numerical experiments, which is closer to the asymptotical value 1.5 and in fact performs better on certain problems than $\alpha = 3.0$.

The limit 300 on the number of optimization iterations is chosen quite arbitrarily, but for the benchmarks in question, we obtain very heterogeneous designs after, say, 50 iterations. Therefore, we believe that averaging over 300 MMA iterations gives quite a good idea of the performance of FETI-DP within the topology optimization context.

Figure 3b provides more insight into the behavior of FETI-DP as the optimization algorithm advances. In this study, we display the number of CG iterations it takes for FETI-DP to converge as the design evolves from the homogeneous material (standard initial guess)

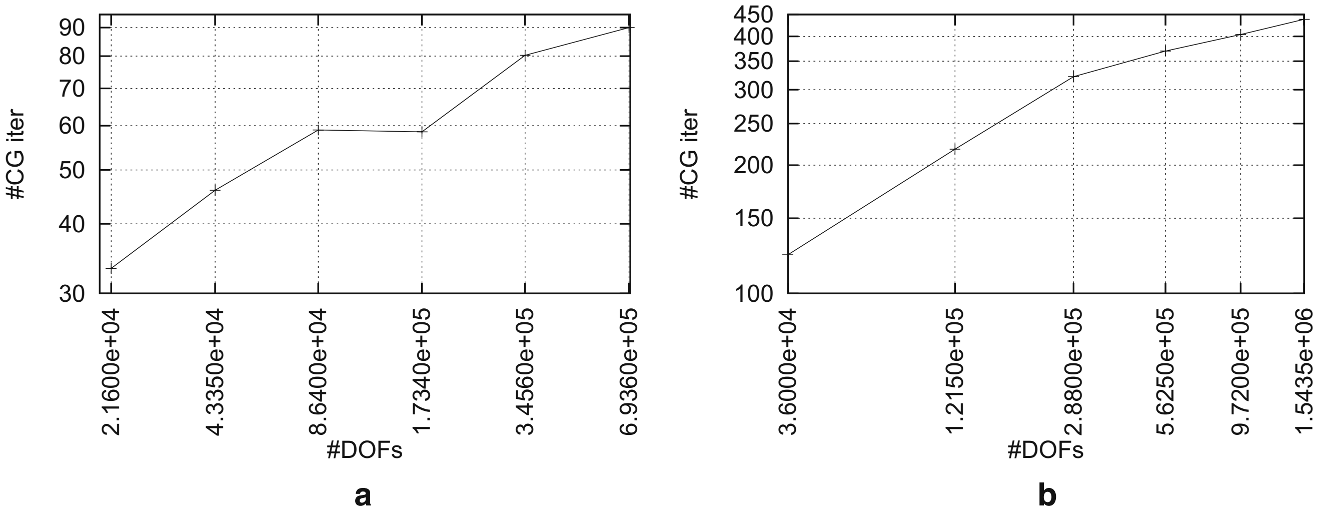

Fig. 4 Numerical scalability test: averaged number of CG iterations over 300 MMA iterations as the number of DOFs increases (and the number of subdomains increases proportionally). **a** 2D MBB beam; **b** 3D MBB beam

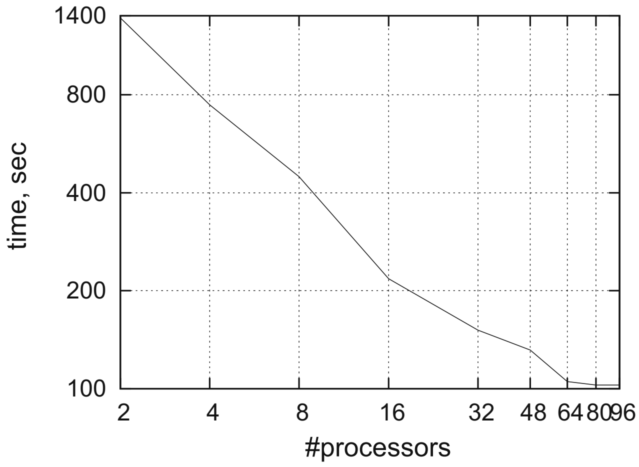

Fig. 5 Parallel scalability test: FETI-DP timings on different number of parallel processors

toward the completely heterogeneous "optimal" design for non-preconditioned ($\alpha = 0.0$) and preconditioned ($\alpha = 2.0$) strategies. One may see the steady increase in the number of CG iterations as the heterogeneity of the subdomains increases and a slight improvement in the performance owing to the additional preconditioning. While an improvement, it is far from keeping the number of iterations nearly constant throughout the optimization run.

5.3 Scalability studies

The next question we want to answer is whether the algorithm practically delivers the constant number of iterations provided we keep the ratio H/h constant [see (13)] in the context of topology optimization, that is, whether it demonstrates numerical scalability in practice. We performed this study, similarly to the previous one, on a 2D and 3D MBB beams, taking an average number of CG iterations over 300 MMA iterations. We kept increasing the number of DOFs and the number of subdomains by the same factor, starting from the mesh 180×60 on 32 subdomains in 2D and $10 \times 20 \times 60$ on 32 subdomains in 3D, with the symmetric Jacobi-type additional preconditioner corresponding to $\alpha = 2.0$. The results are displayed in Fig. 4. One can see that the number of CG iterations most certainly does not remain constant, but at least grows much slower than the number of DOFs or the number of subdomains (approximately as $\#\text{CG iter} \sim 0.08\#\text{DOFs}$ in 2D and $\#\text{CG iter} \sim 0.09\#\text{DOFs}$ in 3D).

A related question is how well the algorithm performs on an increasing number of parallel processors while keeping the problem size, the number of subdomains, and other parameters constant, that is, the parallel scalability. Ideally, the total computing time should be proportional to the inverse of the number of processors, but as some of the time is spent in the communication phase, this ideal behavior is never matched by most practical algorithms. In Fig. 5, we report the time it takes FETI-DP to solve a linear system corresponding to the "optimal" 2D MBB-beam solution (i.e., no optimization) on a 1,020 × 340 regular mesh divided into $1,024$ subdomains. We see that the algorithm speeds up fairly well until the communication time starts to take a significant percentage of the total computational time, which is expected.

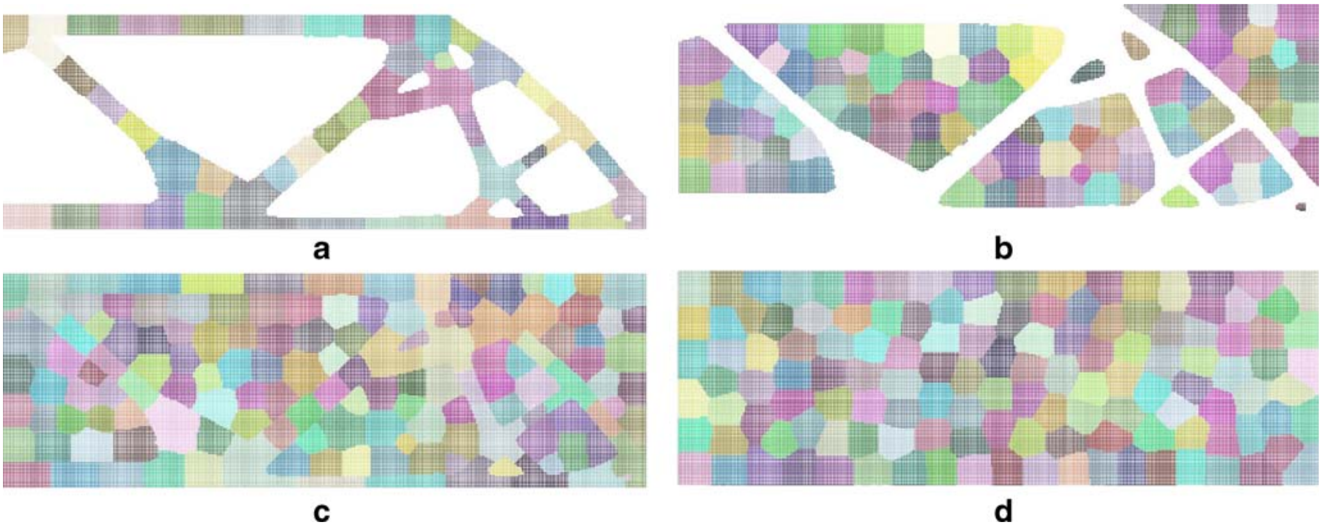

Fig. 6 **a–c** Material-based and **d** optimal decompositions of the domain

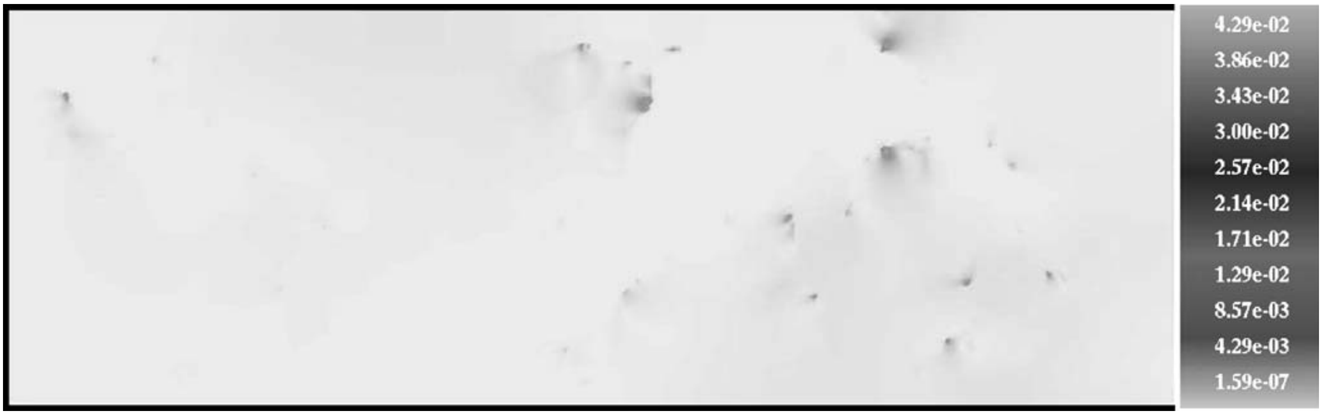

Fig. 7 Study of the unsymmetric Jacobi-type preconditioning: Norm of the difference between the "exact" solution (obtained using the sparse direct solver) and an inexact FETI-DP solution (obtained to the tolerance 10^{-3}) for the "optimal" 2D MBB-beam Fig. 2b and "optimal" domain decomposition Fig. 2c

5.4 Material-based domain decomposition

The numerical scalability of FETI-DP can be maintained even for highly heterogeneous elastic problems *independently* from the difference in elastic properties between "stiff" and "soft" materials as long as they are decomposed into different subdomains (Klawonn et al. 2002). As it is computationally infeasible to maintain the domain decomposition with material decomposition, which changes with every topology optimization iteration, FETI-DP demonstrates less than optimal performance on such problems.

To illustrate the fact that it is the heterogeneity *within* the subdomains that makes FETI-DP converge slowly, we have performed the following test. An "optimal" 2D MBB beam solution, corresponding to the SIMP exponent of 3.0 and the lower density bound of $5.0 \cdot 10^{-2}$ on a 1,020 × 340 regular mesh, has been decomposed into ~1,024 subdomains in two different ways. First, we perform a material-based decomposition; that is, we decomposed the "solid" and the "void" parts of the domain separately Fig. 6a–b and then merged two meshes Fig. 6c. Second, we decomposed the whole domain into subdomains disregarding the difference in material properties, in a way that minimizes the total length of the interface whence the size of the dual system, yet keeps the subdomains geometrically roughly equal and regular (Fig. 6d).

Strikingly, on the material-based decomposition, it takes 20 CG iterations to decrease the residuals below $1 \cdot 10^{-7}$; on the second decomposition, the same goal is reached in 93 CG iterations. These numbers are in accordance with the condition number estimate (14) and indeed demonstrate the efficiency of FETI-DP for most but topology optimization purposes.

5.5 Scaling of the primal residual (asymmetric Jacobi-type preconditioning)

In the hope of improving the performance of FETI-DP for topology optimization, instead of applying the method to a preconditioned system $[\mathsf{D}(\rho)\mathsf{K}(\rho)\mathsf{D}(\rho)[\mathsf{D}^{-1}(\rho)\mathbf{u}] = \mathsf{D}(\rho)f$ as described in Section 5.2, we apply FETI-DP to the original system $\mathsf{K}(\rho)\mathbf{u} = f$, but we measure the primal residual in a different norm:

$$\|\mathsf{D}^{-2}(\rho)[\mathsf{K}(\rho)\mathbf{u} - \mathbf{f}]\|,$$

which essentially corresponds to preconditioning the system from the left only or "ignoring" the primal residual for the "void" nodes in a convergence criterion. Unfortunately, one may see from Fig. 7 that the primal

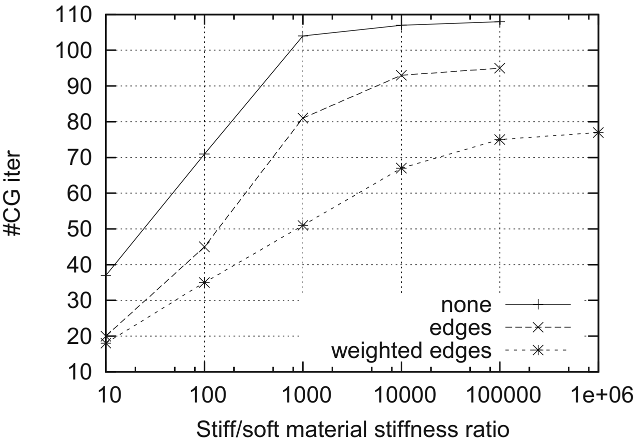

Fig. 8 Increase in the number of CG iterations as the ratio of the Young's moduli of stiff and soft materials increases. *None* no augmentation; *edges* augmentation with translation rigid body modes of edges; *weighted edges* augmentation with stiffness-scaled rigid body modes

error is distributed relatively evenly within the domain and cannot be attributed to void regions only. Thus, this strategy, which works in other applications (Pajot 2006), also does not make a significant improvement in efficiency for topology optimization with FETI-DP.

5.6 Scaling of the augmentation matrix

Finally, we would like to mention one approach that makes a significant difference in the performance of FETI-DP on heterogeneous elastic systems (Dohrmann 2003) but unfortunately is still not enough to help topology optimization. To do this, we must provide more details about FETI-DP than we mentioned in Section 3. One important concept that makes FETI-DP numerically scalable for 3D elasticity problems is that of an augmentation matrix. In addition to the continuity constraints (7), we introduce essentially a duplicate set of *aggregated* constraints (and thus additional dual variables or Lagrange multipliers) as:

$$\mathsf{Q}_b^T \mathsf{B}_r \mathbf{u}_r = \mathbf{0}, \qquad (16)$$

where Q_b is an augmentation matrix. To keep the size of the dual system small, it makes sense to keep the number of columns in the matrix Q_b small. In the standard FETI-DP as introduced in (Farhat et al. 2001), the matrix Q_b is empty; for 3D elliptic problems and forced vibrations problems, it is recommended to set columns of Q_b to translational rigid body modes of each edge (Farhat et al. 2005). That is, for every edge Γ, there are three consecutive columns in the matrix Q_b, namely:

$$\begin{aligned}
\mathbf{q}_x^\Gamma &= (0\ \ldots\ 0\,[1\,0\,0\ \ldots\ 1\,0\,0\ \ldots\ 1\,0\,0]\,0\ \ldots\ 0)^T,\\
\mathbf{q}_y^\Gamma &= (0\ \ldots\ 0\,[0\,1\,0\ \ldots\ 0\,1\,0\ \ldots\ 0\,1\,0]\,0\ \ldots\ 0)^T,\\
\mathbf{q}_z^\Gamma &= (0\ \ldots\ 0\,[0\,0\,1\ \ldots\ 0\,0\,1\ \ldots\ 0\,0\,1]\,0\ \ldots\ 0)^T, \qquad (17)
\end{aligned}$$

where the only non-zero values in the columns are between two square brackets and they correspond to the DOFs associated with the edge Γ. This type of augmentation will be referred to as edge augmentation.

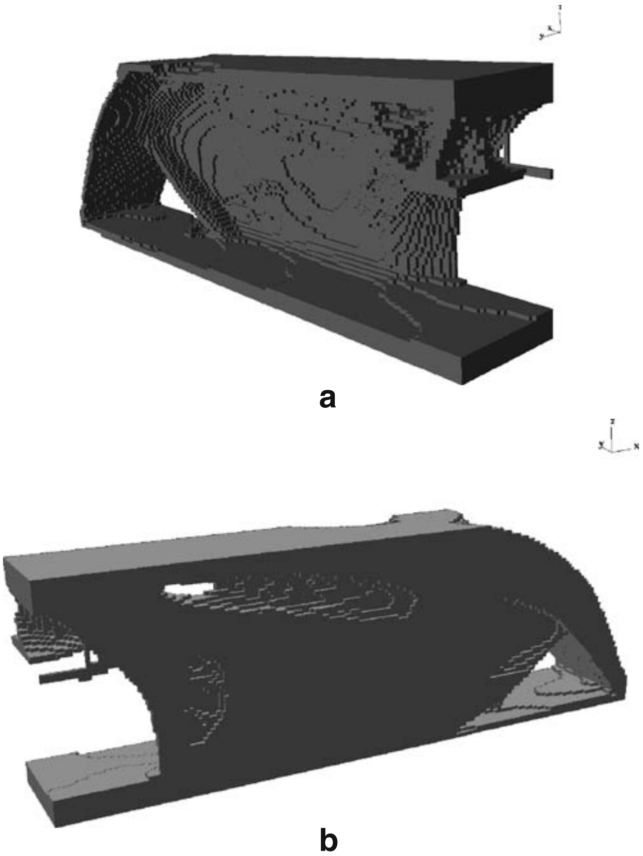

Fig. 9 Compliance minimization under volume constraint: "optimal" 3D MBB-beam

For highly heterogeneous materials, it was suggested (Dohrmann 2003) that the columns of the augmentation matrix should be scaled with the diagonal entries of the stiffness matrix K, corresponding to the DOFs associated with $\mathbf{u}_r$. That is, in (17), instead of entries set to 1, we will have values k_{ii}, where i is the degree of freedom associated with the given entry. This type of the augmentation will be referred to as the weighted edge augmentation.

We studied the performance of FETI-DP with the various augmentation matrices on a pure zero–one "optimal" 2D MBB-beam solution, on a 360 × 120 grid divided into 128 subdomains in an "optimal" way. We keep the Young's modulus of the stiff material fixed and equal to $1.8 \cdot 10^5$ and successively decrease the Young's modulus if the soft (void) material to increase the ill-conditioning of the system.

The results of this study are reported in Fig. 8. The problem with FETI-DP as applied to systems with big jumps in coefficients is that there is a huge difference between the norms of primal and dual residuals. For example, for the 2D MBB example with the ratio of Young's moduli for stiff and soft materials of 10^6, we observe similar ratio in primal and dual residuals. Thus, to solve the (primal) problem to the accuracy of 10^{-7} in terms of the primal residual, one has to solve the corresponding dual interface problem to the precision of 10^{-13}, which is nearly impossible with IEEE double precision floating point arithmetics.

Again, we should mention that, for certain topology optimization problems, it may not be necessary to solve the linear system to a very high accuracy. Nevertheless, we feel that methods working in terms of primal variables yet enjoy the numerical scalability may be more appropriate for topology optimization problems; see, e.g. (Li and Widlund 2006) for an overview.

6 Gallery of examples

In this section, we illustrate the capabilities and the shortcomings of FETI-DP based topology optimization toolbox. We use trilinear Lagrange finite elements (eight-node brick elements) to discretize all numerical examples in this section.

One of the first examples that we have solved is a 3D MBB beam, see Section 5.1 and Fig. 2d. We used a regular 30 × 60 × 180 mesh, "optimally" decomposed into 868 subdomains. This example is rather small, it contains only about $9.72 \cdot 10^5$ DOFs and $3.24 \cdot 10^5$ optimization variables. We use standard filtering (Bourdin 2001; Sigmund 1997; Sigmund and Petersson 1998) with a filter radius of 1.2 times the size of the finite element to prevent checkerboarding. A "typical" function evaluation for this example requires 300 CG iterations within FETI-DP, which takes approximately 85 s on 96 processors on our cluster. This example is "good" for FETI-DP because it contains only an integral energy-type objective function, which is very important for residual minimization-driven iterative linear solvers, such as FETI-DP. A couple of views of an "optimal" design that we have obtained after 122 MMA iterations (with the maximal step size restricted to 0.1) is shown in Fig. 9. FETI-DP in this example demonstrated the same behavior as described in Section 5; that is, the number of CG iterations for the interface problems increases with the increasing heterogeneity of the solution as the optimization algorithm progresses.

The next example is a design of a compliant mechanism. We design a 3D force inverter, as shown in

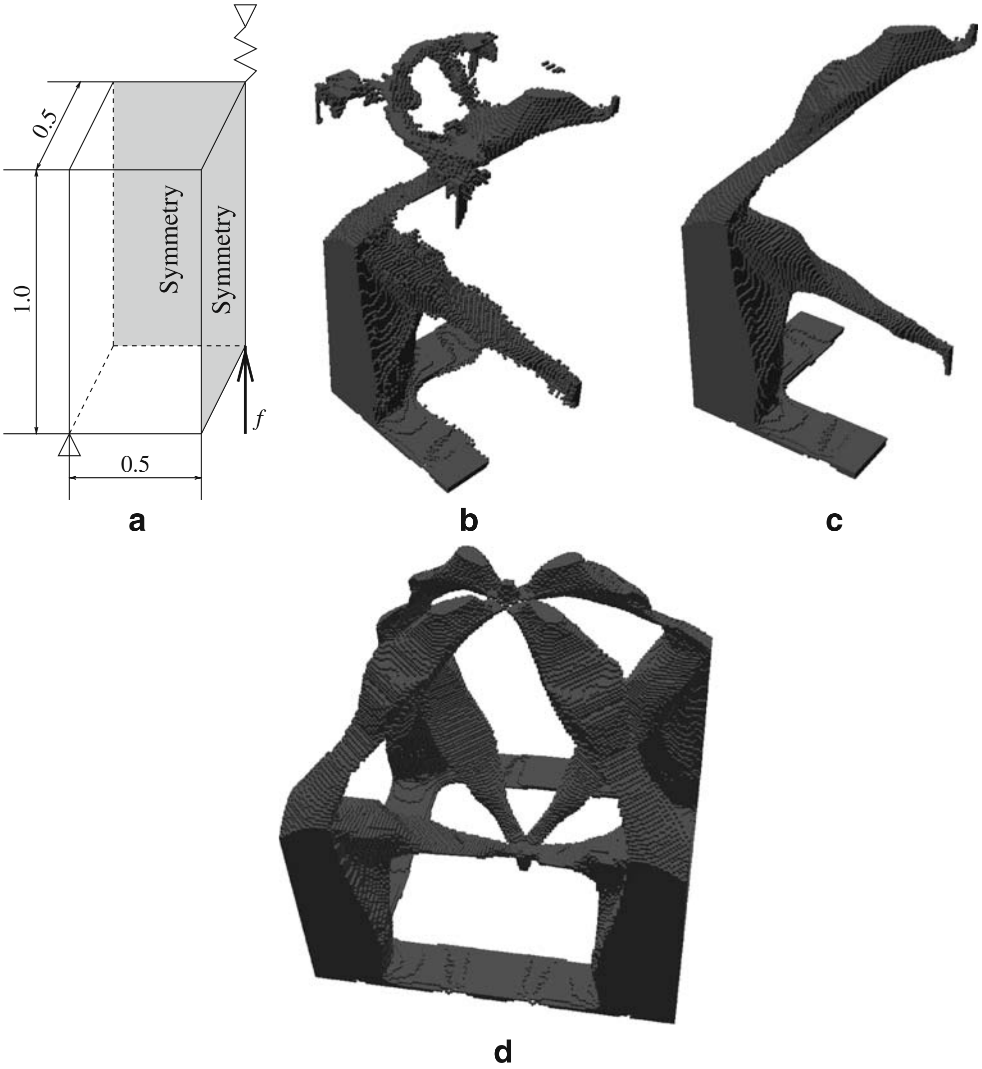

Fig. 10 Optimization of a compliant mechanism (inverter). (a) Problem setup; (b) "optimal" design without post-processing; (c) post-processed "optimal" design; (d) post-processed "optimal" design including symmetries

Fig. 10a, where a force is applied in an upward direction at one of the vertices of the design “box,” and we are interested in maximizing the downward displacement of a spring-supported adjacent vertex. The amount of material available is limited to 20% of the volume of the design domain, and a bound on the vertical displacement of the loaded point is imposed. We use SIMP and filtering as in the previous example (3D MBB beam). This particular design problem requires much more precision from the linear solver, as we need individual nodal displacements and their derivatives to evaluate both the objective function and the constraints. This example is also a little bit larger in terms of number of DOFs. We use a grid of 75 × 75 × 150 nodes (approximately $2.5 \cdot 10^6$ DOFs, $8.4 \cdot 10^5$ optimization variables) divided into approximately 1,024 subdomains. We use a filter with a radius of 2 times the size of the finite element to prevent checkerboarding. We solved this example on 8 processors, with about 360 CG iterations per function evaluation taking approximately 830 s. With FETI-DP settings similar to the previous example, after 68 MMA iterations (with the maximum step-size set to 0.1), we obtain a design shown in Fig. 10b. While this design is definitely feasible and performs the function of a force inverter, it can certainly be improved upon; in particular, it contains a lot of numerical “noise” coming from imprecise evaluations of the gradients. Figure 10 c and d contains the same design with the “noise” removed after some post-processing.

The last example is a design of an elastic surface wave-guide (Rupp et al. 2006). In this study, we try to guide a 2-GHz harmonic load entering the design domain at an input port toward an output port (cf. Fig. 11a) by distributing aluminum within a thin layer of silicon; the whole structure is attached to a silicon substrate at the bottom. We surround the domain with non-reflecting boundary conditions on all sides except the top, thus modeling 3D halfspace. The non-reflecting boundary conditions are modeled with viscous damping elements. For manufacturing purposes, we do not allow the design to change in the vertical direction, so that etching or lithography may be used to build it. This example was discretized with a 160 × 160 × 64 elements, resulting in approximately $5.1 \cdot 10^6$ complex, or 10^7 real, DOFs. At the same time, the optimization problem has only $2.6 \cdot 10^4$ variables owing to the manufacturability considerations. On 12 processors, 225 generalized conjugate residual iterations needed for a typical function evaluation require approximately 2,240 s, this number dropping to approximately 510 s on 80 processors. No filtering or penalization is used or usually needed to obtain well-defined 0–1 “optimal” designs in elastic wave propagation (Rupp et al. 2006). The design shown in Fig. 11b is obtained after 36 MMA iterations (with the maximum step-size set to 0.25), and it still contains

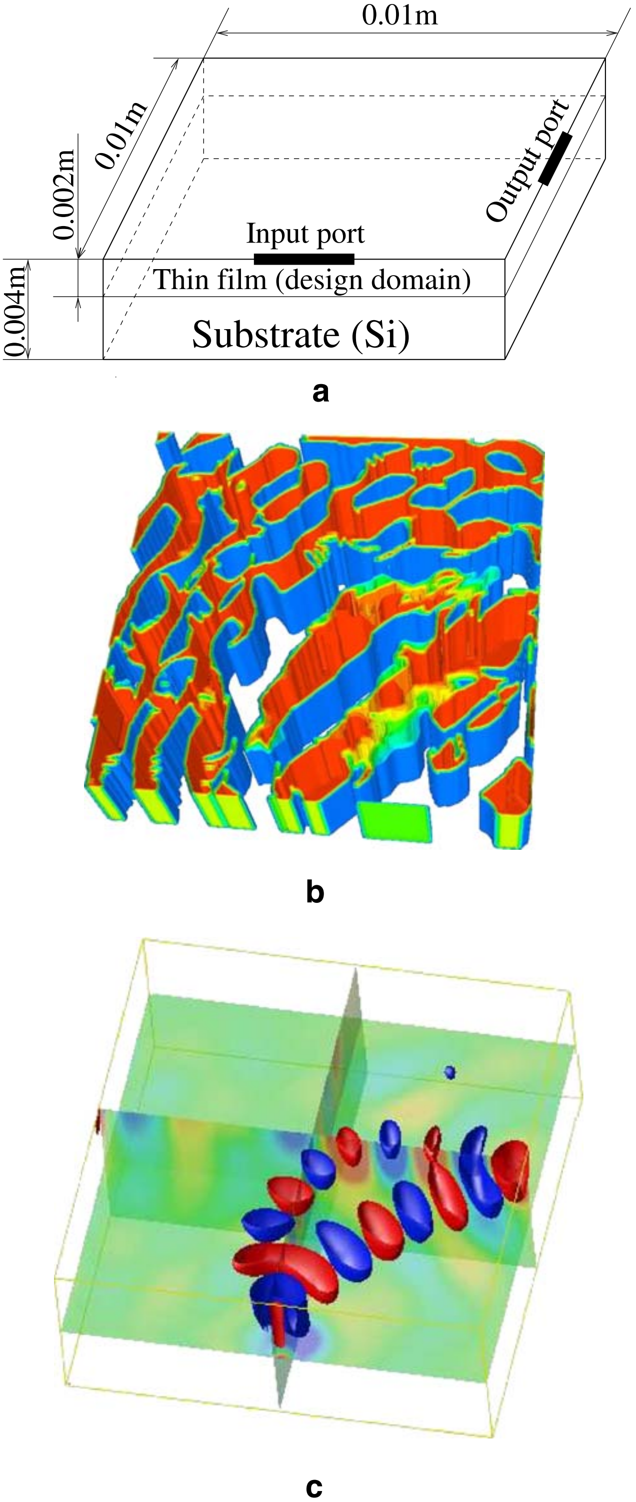

Fig. 11 Design of an elastic surface wave-guide (“wave-bender”). **a** Problem setup; **b** “optimal” design; **c** surface wave propagation corresponding to design (**b**)

some intermediate density areas (around 8% of optimization variables are strictly between 0 and 1), which can be fixed by simplistic post-processing techniques without altering the objective function much, or by letting the optimization algorithm proceed further (Rupp et al. 2006). The elastic surface wave-field corresponding to the "optimal" design is shown in Fig. 11c, and one can see that the majority of the elastic energy is indeed guided to the output port.

Numerically, this example is very convenient for FETI-DP; elastic properties of silicon and aluminum are not as different as the typical stiff and soft materials in compliance minimization problems with SIMP. However, the linear system is not definite any more, and therefore, there is no theoretical numerical scalability estimate, such as (13), available. In addition, the solver seems to have difficulties solving the linear system if the viscous damping elements are replaced by perfectly matching layer elements, where the material properties vary very fast from one element to another (see, e.g., Basu and Chopra 2003).

7 Concluding remarks and future research directions

In this paper, we studied the behavior of the state-of-the-art scalable linear solver FETI-DP for the purposes of topology optimization. FETI-DP is numerically scalable for problems with large jumps in coefficients, uniformly in the size of the jump, as long as the subdomains are kept homogeneous, which is not the case in the topology optimization. With numerical examples, we demonstrated that the performance of FETI-DP deteriorates quickly as the heterogeneity of the subdomains increases or as the optimization algorithm progresses from the initial homogeneous design toward "optimal" 0–1 designs.

For certain problems, where jumps in the coefficients are not so large, such as in the case with topology optimization problems in elastic wave propagation, the method can be recommended. For "classical" solid–void topology optimization problems in linearized elasticity with SIMP, the dual-primal method has to confront the difficulty that the ratio of the primal and dual residuals is huge. Therefore, we expect that domain decomposition techniques operating in terms of primal variables and primal residuals only, such as balancing domain decomposition by constraints (Dohrmann 2003; Mandel et al 2005), may demonstrate better performance, as they effectively avoid the latter difficulty.

Acknowledgements We thank Charbel Farhat for providing us with an implementation of the FETI-DP algorithm and Radek Tezaur for his help working with this implementation. We are also grateful to Kendall H. Pierson for the discussion on improving the performance of FETI-DP and for the reference (Dohrmann 2003) and to the anonymous reviewers for their insightful comments helping us to improve the presentation of the material.

References

Basu U, Chopra AK (2003) Perfectly matched layers for time-harmonic elastodynamics of unbounded domains: theory and finite element implementation. Comput Methods Appl Mech Eng 192(11–12):1337–1375

Bendsøe MP (2006) Multidisciplinary topology optimization. In: Proc. 11th AIAA/ISSMO Symposium on Multidiciplinary Analysis and Optimization

Bendsøe MP, Kikuchi N (1988) Generating optimal topologies in structural design using a homogenization method. Comput Methods Appl Mech Eng 71(2):197–224

Bendsøe MP, Sigmund O (2003) Topology optimization: theory, methods, and applications. Springer, Berlin

Borrvall T, Petersson J (2001) Topology optimization using regularized intermediate density control. Comput Methods Appl Mech Eng 190(37–38):4911–4928

Bourdin B (2001) Filters in topology optimization. Int J Numer Methods Eng 50(9):2143–2158

Bramble JH, Pasciak JE, Schatz AH (1986) The construction of preconditioners for elliptic problems by substructuring. I. Math Comp 47(175):103–134

Dohrmann CR (2003) A study of two domain decomposition preconditioners. Tech. Rep. SAND2003-4391, Sandia National Laboratories, Albuquerque, New Mexico 87185 and Livermore, California 94550

Duysinx P, Bruyneel M (2002) Recent progress in preliminary design of mechanical components with topology optimisation. In: Chedmail P, Cognet G, Fortin C, Mascle C, Pegna J (eds) Book of selected papers presented at 3rd Conference on Integrated Design and Manufacturing in Mechanical Engineering IDMME2000/Forum 2000 of SCGM/CSME, Kluwer Publ.

Evgrafov A, Pingen G, Maute K (2006) Topology optimization of fluid problems by the lattice Boltzmann method. In: Bendsøe MP, Olhoff N, Sigmund O (eds) IUTAM Symposium on Topological Design Optimization of Structures, Machines and Materials: Status and Perspectives. Springer, Netherlands, pp 559–568

Farhat C, Lesoinne M, LeTallec P, Pierson K, Rixen D (2001) FETI-DP: a dual–primal unified FETI method—part I: a faster alternative to the two-level FETI method. Int J Numer Methods Eng 50(7):1523–1544

Farhat C, Li J, Avery P (2005) A FETI-DP method for the parallel iterative solution of indefinite and complex-valued solid and shell vibration problems. International Journal for Numerical Methods in Engineering 63(3):398–427

Kim TS, Kim JE, Kim YY (2004) Parallelized structural topology optimization for eigenvalue problems. Int J Solids Struct 41:2623–2641

Klawonn A, Widlund OB, Dryja M (2002) Dual–primal FETI methods for three-dimensional elliptic problems with heterogeneous coefficients. SIAM J Numer Anal 40(1): 159–179 (electronic)

Lee FH, Phoon K, Lim K, Chan S (2002) Performance of Jacobi preconditioning in Krylov subspace solution of finite element equations. Int J Numer Anal Meth Geomech 26:341–372

Li J, Widlund OB (2006) FETI-DP, BDDC, and block Cholesky methods. Int J Numer Methods Eng 66(2):250–271

Mahdavi A, Balaji R, Frecker M, Mockensturm E (2006) Topology optimization of 2d continua for minimum compliance using parallel computing. Struct Multidisc Optim 32(2): 121–132

Mandel J, Tezaur R (2001) On the convergence of a dual-primal substructuring method. Numer Math 88(3):543–558

Mandel J, Dohrmann CR, Tezaur R (2005) An algebraic theory for primal and dual substructuring methods by constraints. Appl Numer Math 54(2):167–193

O'Neal D, Murgie S (2001) ANSYS benchmarking project: evaluation of the distributed domain solver. Tech. Rep., ANSYS, Inc., Pittsburgh, PA, USA

Pajot JM (2006) Topology optimization of geometrically nonlinear structures including thermo-mechanical coupling. PhD thesis, University of Colorado at Boulder

Petersson J (1999) A finite element analysis of optimal variable thickness sheets. SIAM J Numer Anal 36(6):1759–1778

Rockafellar RT, Wets RJB (1998) Variational analysis. Springer, Berlin

Rozvany GIN, Zhou M, Birker T (1992) Generalized shape optimization without homogenization. Struct Optim 4:250–254

Rupp CJ, Evgrafov A, Maute K, Dunn ML (2006) Design of phononic materials/structures for surface wave devices using topology optimization. Struct Multidisc Optim Online, doi:10.1007/s00158-006-0076-0

Saad Y (2003) Iterative methods for sparse linear systems, 2nd edn. Society for Industrial and Applied Mathematics, Philadelphia, PA

Sigmund O (1997) On the design of compliant mechanisms using topology optimization. Mech Struct Mach 25(4): 493–524

Sigmund O (2001) A 99 line topology optimization code written in MATLAB. Struct Multidisc Optim (2):120–127

Sigmund O, Petersson J (1998) Numerical instabilities in topology optimization: a survey on procedures dealing with checkerboards, mesh-dependencies and local minima. Struct Multidisc Optim 16(1):68–75

Svanberg K (1987) The method of moving asymptotes—a new method for structural optimization. Int J Numer Methods Eng 24(2):359–373

Svanberg K (2002) A class of globally convergent optimization methods based on conservative convex separable approximations. SIAM J Optim 12(2):555–573

Vemaganti K, Lawrence EW (2005) Parallel methods for optimality criteria-based topology optimization. Comp Methods Appl Mech Eng 194:3637–3667

Wang S, de Sturler E, Paulino GH (2007) Large-scale topology optimization using preconditioned Krylov subspace methods with recycling. Int J Numer Methods Eng 69(12): 2441–2468