Struct Multidisc Optim (2016) 53:1315–1333
DOI 10.1007/s00158-015-1381-2

RESEARCH PAPER

An efficient second-order SQP method for structural topology optimization

Susana Rojas-Labanda[1] · **Mathias Stolpe**[1]

Received: 24 April 2015 / Revised: 3 November 2015 / Accepted: 29 November 2015 / Published online: 10 March 2016

Abstract This article presents a Sequential Quadratic Programming (SQP) solver for structural topology optimization problems named TopSQP. The implementation is based on the general SQP method proposed in Morales et al. (J Numer Anal 32(2):553–579; 2010) called SQP+. The topology optimization problem is modelled using a density approach and thus, is classified as a nonconvex problem. More specifically, the SQP method is designed for the classical minimum compliance problem with a constraint on the volume of the structure. The sub-problems are defined using second-order information. They are reformulated using the specific mathematical properties of the problem to significantly improve the efficiency of the solver. The performance of the TopSQP solver is compared to the special-purpose structural optimization method, the Globally Convergent Method of Moving Asymptotes (GCMMA) and the two general nonlinear solvers IPOPT and SNOPT. Numerical experiments on a large set of benchmark problems show good performance of TopSQP in terms of number of function evaluations. In addition, the use of second-order information helps to decrease the objective function value.

This research is funded by the Villum Foundation through the research project *Topology Optimization – the Next Generation (NextTop)*.

✉ Susana Rojas-Labanda
srla@dtu.dk

Mathias Stolpe
matst@dtu.dk

[1] Department of Wind Energy, Technical University of Denmark, Frederiksborgvej 399, 4000 Roskilde, Denmark

Keywords Topology optimization · Sequential Quadratic Programming · Minimum compliance · Second-order method · Hessian approximation

1 Introduction

Structural topology optimization determines the design in a domain by minimizing an objective function under certain constraints, for a given set of boundary conditions and loads. It is common to minimize compliance or volume subject to limitations on the displacements, volume, or stresses. Topology optimization is a mathematical approach where the design domain is often discretized using finite elements for design parametrization and structural analysis. More details of these problems can be found in the monograph Bendsøe and Sigmund (2003).

Several special purpose methods have been implemented to solve structural topology optimization problems. Examples include the Method of Moving Asymptotes, (MMA) (Svanberg 1987), its globally convergent version, (GCMMA) (Svanberg 2002), and the Convex Linearization (CONLIN) method (Fleury 1989a). These first-order methods solve a sequence of convex sub-problems based on separable approximations of the objective and constraint functions. Different variations of MMA using the diagonal of the second-order derivatives are proposed in Fleury (1989b, c), evaluating the benefits of using partial second-order information.

Although it is not very commonly reported in the literature, general nonlinear optimization solvers are also applicable to topology optimization problems. The numerical experiments in Rojas-Labanda and Stolpe (2015) show that general purpose solvers, such as interior-point and sequential quadratic programming methods, can be used for

solving these type of problems. In these numerical experiments, SNOPT (an SQP method) (Gill et al. 2005) requires very few iterations to converge to a Karush-Kuhn-Tucker (KKT) point. In addition, second-order information helps to produce designs with low objective function value as demonstrated by the interior-point method IPOPT (Wächter and Biegler 2006). The results of the benchmarking study in Rojas-Labanda and Stolpe (2015) have motivated the implementation of a second-order SQP method for structural topology optimization problems called TopSQP. Due to the use of second-order information, TopSQP is expected to converge faster than first-order methods. In addition, the authors expect that TopSQP produces designs with a compliance lower than first-order methods.

In the SQP family of methods, there are numerous variations of algorithms, but all of them are characterized by the same idea. They find approximate solutions to a sequence of normally convex sub-problems. A quadratic objective function models the Lagrangian while the original constraints are linearized. For general information of SQP see e.g. Boggs and Tolle (1995) and Gill and Wong (2012).

One of the main properties of SQP methods is the fast convergence when close to a KKT point. However, the performance of SQP depends, in general, on the starting point. It is quite difficult to use second-order information (see Section 2) when the problems are nonconvex. In practice, most of the SQP algorithms use quasi-Newton approximations of the Hessian in order to obtain a convex sub-problem, see e.g. Gill and Wong (2012).

The main differences between SQP algorithms are how the search direction is computed, how the search direction is accepted or rejected, and how inequality constraints are dealt with. Both line search (Moré and Thuente 1994) and trust region strategies (Conn et al. 1987) can be applied in SQP solvers. Additionally, to ensure convergence, merit functions (Boggs and Tolle 1984) or filter methods (Fletcher and Leyffer 2002) are used. Regarding how the method obtains the active inequality constraints, SQP is classified as equality constrained quadratic programming (EQP) or inequality constrained quadratic programming (IQP), see e.g. Gould and Robinson (2010). Finally, there are different ways to approximate the Hessian of the Lagrangian, using either limited-memory approximations like BFGS (Broyden - Fletcher - Goldfarb - Shanno) (Dennis and Moré 1977) or some information of the Hessian. Examples of recent SQP algorithms can be found in Gill and Wong (2012), Gill and Robinson (2013), Gould and Robinson (2010), Morales et al. (2010), Hager (1999), Curtis et al. (2014), and Shen et al. (2010), among others.

There are several software in the optimization community based on SQP methods. For instance, SNOPT (Gill et al. 2005), the NLPQLP solver (Schittkowski 2002), and NPSOL (Gill et al. 1998), where a line search is combined with different penalty functions, the trust region with a filter method in FilterSQP (Fletcher and Leyffer 1998), or the new SQP implemented in KNITRO (Byrd et al. 1996), among others. More details of nonlinear solvers can be found in e.g. Leyffer and Mahajan (2010). Nevertheless, the use of SQP methods in topology optimization is seemingly not very popular, and very few references have been found in this regard, see e.g. Orozco and Ghattas (1997), Dreyer et al. (2000), and Etman et al. (2012).

This article contains a detailed description of an efficient sequential quadratic programming method for maximum stiffness structural topology optimization problems. The foundation is the SQP+ method introduced in Morales et al. (2010). SQP+ contains two phases, the inequality quadratic phase (IQP), where an inequality constrained convex quadratic sub-problem is solved. Then, an equality constrained quadratic phase (EQP), where the active constraints found for the IQP are used. The step generation is done using a line search strategy in conjunction with a reduction in a merit function. In the special-purpose implementation TopSQP, the IQP phase uses second-order information to define a convex approximation of the Hessian instead of using the traditional BFGS. To compare the performance of TopSQP, an SQP+ is implemented in which a limited-memory BFGS is used for the IQP phase. Based on the specific structure of the problem formulation, both phases are efficiently reformulated to reduce computational cost. The reformulations avoid one of the most expensive steps in topology optimization problems, which is the computation of the inverse of the stiffness matrix (involved in the Hessian of compliance).

The performance of the proposed TopSQP is compared to a particular implementation of SQP+, the specific purpose GCMMA, and two state-of-the-art software for numerical optimization SNOPT and IPOPT. The comparative study is done using performance profiles (Dolan and Moré 2002) on a test set of 225 medium-size minimum compliance problem instances described in Rojas-Labanda and Stolpe (2015).

The paper is organized as follows. Section 2 introduces the SQP+ algorithm proposed in Morales et al. (2010) for a general nonlinear problem and Section 3 briefly defines the topology optimization problem under consideration. Then, the Hessian of the Lagrangian function and some possible convex approximations are proposed in Section 4. The efficient reformulations of the IQP and EQP phases of TopSQP are gathered in Sections 5 and 6, respectively. Some implementation details are collected in Section 7. The comparative study of the performance of TopSQP is reported in Section 8, followed by a list of the main limitations this new

algorithm may have for topology optimization problems, in Section 9. Finally, Section 10 draws the main conclusions from the results and outlines recommendations for the future work.

2 Sequential quadratic programming with an additional equality constrained phase

A general sequential quadratic programming method generates approximate solutions using a quadratic model of the Lagrangian function and linearization of the constraints.

The general nonlinear constrained problem under consideration is

$$\begin{aligned} \underset{\mathbf{x}\in\mathbb{R}^n}{\text{minimize}} \quad & f(\mathbf{x}) \\ \text{subject to} \quad & g_i(\mathbf{x}) \leq 0 \quad i = 1,\ldots,m, \\ & l_i \leq x_i \leq u_i \;\; i = 1,\ldots,n, \end{aligned} \tag{NLP}$$

where $f : \mathbb{R}^n \longrightarrow \mathbb{R}$ and $g_i : \mathbb{R}^n \longrightarrow \mathbb{R}$ are assumed to be twice continuously differentiable functions.

A conventional SQP method approximates (NLP) at a given iterate $\mathbf{x}_k$, and uses the solution of the quadratic sub-problem to produce a search direction $\mathbf{d}_k$. The solver ensures convergence to a KKT point by enforcing, for instance, an improvement in a merit function. This class of methods has shown fast local convergence (see e.g. Nocedal and Wright (1999) and Morales et al. (2010)), however, the theoretical properties do not hold when the Hessian is indefinite, producing some difficulties to the solver. Since nonconvex optimization problems are NP-hard problems (Murty and Kabadi 1987; Manyem and Ugon 2012), a minimizer of the sub-problem does not guarantee the convergence of the algorithm, see e.g. Prieto (1989).

The sequential quadratic programming method proposed in this article is based on the algorithm SQP+ in Morales et al. (2010). SQP+ attempts to improve the convergence rate by including two phases. First, an inequality constrained quadratic convex sub-problem is solved. Second, the set of active constraints is estimated, and with them, an EQP sub-problem is defined and solved, ignoring the rest of the constraints. The EQP estimated solution refines the search direction, producing fast convergence. Additionally, the IQP iterate of the proposed TopSQP is expected to be more precise than in Morales et al. (2010) since a positive definite approximation based on the indefinite Hessian is used instead of the BFGS approach. The approximation is based on second-order information (see Section 4).

Throughout this section, the SQP+ method, summarized in Algorithm 1, is outlined for the general nonlinear problem (NLP).

Algorithm 1 SQP+ algorithm from Morales et al. (2010).

Require: Define the starting point $\mathbf{x}_0$, the initial Lagrangian multipliers λ_0, ξ_0, η_0 and the optimality tolerance ω.

Set $\sigma = 10^{-4}, \kappa = 0.5, k = 1, \pi = 1$.
repeat
  Define an approximation of the Hessian of the Lagrangian function, $\mathbf{B}_k \succ 0$ such as $\mathbf{B}_k \approx \nabla^2\mathcal{L}(\mathbf{x}_k, \lambda_k)$.
  Solve the IQP sub-problem as explained in Section 2.2 where the solution is $(\mathbf{d}_k^{iq}, \lambda_k^{iq}, \xi_k^{iq}, \eta_k^{iq})$.
  Determine the working set of active constraints, $\mathcal{W}_k^g$ and $\mathcal{W}_k^b = \{\mathcal{W}_k^u \cup \mathcal{W}_k^l\}$, defined in Section 2.3.
  Solve the EQP sub-problem as explained in Section 2.3 where the solution is $(\mathbf{d}_k^{eq}\ \lambda_k^{eq}\ \xi_k^{eq}\ \eta_k^{eq})$.
  Compute the contraction parameter $\beta \in (0, 1]$ such as the linearized contraints of the IQP sub-problem are feasible at the iterate point $\mathbf{x}_k + \mathbf{d}_k^{iq} + \beta\mathbf{d}_k^{eq}$.
  Acceptance/rejection step. Use of line search strategy:
  if $\phi_\pi(\mathbf{x}_k + \mathbf{d}_k^{iq} + \beta\mathbf{d}_k^{eq}) \leq \phi_\pi(\mathbf{x}_k) - \sigma qred_\pi(\mathbf{d}_k^{iq})$
    $\mathbf{d}_k = \mathbf{d}_k^{iq} + \beta\mathbf{d}_k^{eq}$.
  else
    Find $\alpha = \{1, \kappa, \kappa^2, ...\}$ such that $\phi_\pi(\mathbf{x}_k + \alpha\mathbf{d}_k^{iq}) \leq \phi_\pi(\mathbf{x}_k) - \sigma\alpha qred_\pi(\mathbf{d}_k^{iq})$.
    $\mathbf{d}_k = \alpha\mathbf{d}_k^{iq}$.
  endif
  Update the primal iterate $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{d}_k$.
  Update the Lagrangian multiplier estimates λ_{k+1}, ξ_{k+1}, η_{k+1} with the strategy explained in Section 2.5.
  Update the penalty parameter π.
  Compute the ∞-norm of KKT conditions of the original problem (NLP).
  Set $k \leftarrow k + 1$.
until `convergence`
return

2.1 Optimality conditions

The Lagrangian function of (NLP) is defined as

$$\begin{aligned} \mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\xi}, \boldsymbol{\eta}) = f(\mathbf{x}) &+ \sum_{i=1}^{m} \lambda_i g_i(\mathbf{x}) \\ &+ \sum_{i=1}^{n} \xi_i(x_i - u_i) + \sum_{i=1}^{n} \eta_i(l_i - x_i). \end{aligned}$$

Here $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_m)^T$, $\boldsymbol{\xi} = (\xi_1, \ldots, \xi_n)^T$, and $\boldsymbol{\eta} = (\eta_1, \ldots, \eta_n)^T$ are the Lagrangian multipliers of the inequality,

the upper bound, and the lower bound constraints, respectively.

The first-order necessary conditions for a primal-dual point $(\overline{\mathbf{x}}, \overline{\boldsymbol{\lambda}}, \overline{\boldsymbol{\xi}}, \overline{\boldsymbol{\eta}})$ to be a local optimal solution of the problem (NLP) are gathered in the Karush-Kuhn-Tucker (KKT) conditions (1)-(9), see e.g. Nocedal and Wright (1999).

$$\nabla \mathcal{L}(\overline{\mathbf{x}}, \overline{\boldsymbol{\lambda}}, \overline{\boldsymbol{\xi}}, \overline{\boldsymbol{\eta}}) = \nabla f(\overline{\mathbf{x}}) + J(\overline{\mathbf{x}})^T \overline{\boldsymbol{\lambda}} + \overline{\boldsymbol{\xi}} - \overline{\boldsymbol{\eta}} = \mathbf{0}, \quad (1)$$

$$g_i(\overline{\mathbf{x}}) \leq 0 \quad i = 1, \ldots, m, \quad (2)$$

$$l_i \leq \overline{x}_i \leq u_i \quad i = 1, \ldots, n, \quad (3)$$

$$\overline{\lambda}_i \geq 0 \quad i = 1, \ldots, m, \quad (4)$$

$$\overline{\xi}_i \geq 0 \quad i = 1, \ldots, n, \quad (5)$$

$$\overline{\eta}_i \geq 0 \quad i = 1, \ldots, n, \quad (6)$$

$$g_i(\overline{\mathbf{x}})\overline{\lambda}_i = 0 \quad i = 1, \ldots, m, \quad (7)$$

$$(\overline{x}_i - u_i)\overline{\xi}_i = 0 \quad i = 1, \ldots, n, \quad (8)$$

$$(l_i - \overline{x}_i)\overline{\eta}_i = 0 \quad i = 1, \ldots, n. \quad (9)$$

Here, $J(\mathbf{x}) = [\nabla g_i(\mathbf{x})^T]_{i=1,\ldots,m} : \mathbb{R}^n \mapsto \mathbb{R}^{m \times n}$ is the Jacobian of the inequality constraints. Equation (1) refers to the stationarity condition, (2)-(3) are the primal feasibility conditions, and (7)-(9) are the complementarity conditions. In addition, some Constraint Qualification (CQ) must hold at $\overline{\mathbf{x}}$, see Nocedal and Wright (1999) and Luenberger and Ye (2008).

In practice, SQP+ considers that $\overline{\mathbf{x}}$ is an optimal solution if the stationarity, feasibility, and complementarity conditions are satisfied within some positive tolerance, i.e.

$$\left\| \nabla f(\overline{\mathbf{x}}) + J(\overline{\mathbf{x}})^T \overline{\boldsymbol{\lambda}} + \overline{\boldsymbol{\xi}} - \overline{\boldsymbol{\eta}} \right\|_\infty \leq \epsilon_1,$$

$$\left\| \begin{bmatrix} \mathbf{g}(\overline{\mathbf{x}})^+ \\ \mathbf{g}_u(\overline{\mathbf{x}})^- \\ \mathbf{g}_l(\overline{\mathbf{x}})^- \end{bmatrix} \right\|_\infty \leq \epsilon_2,$$

$$\left\| \begin{bmatrix} \mathbf{h}_g(\overline{\mathbf{x}}, \overline{\boldsymbol{\lambda}}) \\ \mathbf{h}_u(\overline{\mathbf{x}}, \overline{\boldsymbol{\xi}}) \\ \mathbf{h}_l(\overline{\mathbf{x}}, \overline{\boldsymbol{\eta}}) \end{bmatrix} \right\|_\infty \leq \epsilon_3,$$

for some given constants $\epsilon_1 > 0$, $\epsilon_2 > 0$ and $\epsilon_3 > 0$. Here,

$$\mathbf{g}(\mathbf{x})^+ \triangleq [\max\{0, g_i(\mathbf{x})\}]_{i=1,\ldots,m},$$
$$\mathbf{g}_u(\mathbf{x})^- \triangleq [\max\{0, -(u_i - x_i)\}]_{i=1,\ldots,n},$$
$$\mathbf{g}_l(\mathbf{x})^- \triangleq [\max\{0, -(x_i - l_i)\}]_{i=1,\ldots,n},$$
$$\mathbf{h}_g(\mathbf{x}, \boldsymbol{\lambda}) \triangleq \left[g_i(\mathbf{x})\lambda_i \right]_{i=1,\ldots,m},$$
$$\mathbf{h}_u(\mathbf{x}, \boldsymbol{\xi}) \triangleq \left[(x_i - u_i)\xi_i \right]_{i=1,\ldots,n},$$
$$\mathbf{h}_l(\mathbf{x}, \boldsymbol{\eta}) \triangleq \left[(l_i - x_i)\eta_i \right]_{i=1,\ldots,n}.$$

2.2 Solving the IQP sub-problem

The inequality constrained quadratic phase (IQP) approximates the problem (NLP) with a convex quadratic model of the Lagrangian function and a linearization of the constraints. Thus, a positive definite matrix, $\mathbf{B}_k$, computed in Step 3 of Algorithm 1 is crucial to define the IQP problem.

$$\begin{aligned} \underset{\mathbf{d} \in \mathbb{R}^n}{\text{minimize}} \quad & \nabla f(\mathbf{x}_k)^T \mathbf{d} + \tfrac{1}{2} \mathbf{d}^T \mathbf{B}_k \mathbf{d} \\ \text{subject to} \quad & g(\mathbf{x}_k) + J(\mathbf{x}_k)\mathbf{d} \leq \mathbf{0}, \\ & \tilde{l}_i \leq d_i \leq \tilde{u}_i \qquad i = 1, \ldots, n. \end{aligned} \quad \text{(IQP)}$$

The lower and the upper bounds of d_i are defined with $\tilde{l}_i = l_i - (\mathbf{x}_k)_i$ and $\tilde{u}_i = u_i - (\mathbf{x}_k)_i$, $i = 1, \ldots, n$, respectively.

From now on, the linearization of the constraints in (IQP) is assumed to result in a feasible problem (see Section 3). The primal-dual solution of the sub-problem $(\mathbf{d}_k^{iq}, \boldsymbol{\lambda}_k^{iq}, \boldsymbol{\xi}_k^{iq}, \boldsymbol{\eta}_k^{iq})$, is used to estimate first, the set of active constraints and second, the search direction.

The most important characteristic of this IQP phase is the convexity of the optimization problem. Any local optimal solution of a convex problem is a global solution. Furthermore, for the IQP in this work, existence of solution is ensured (see Section 4). In addition, the problem always has a descent direction until convergence (see e.g. Boyd and Vandenberghe (2010)).

The proposed IQP (in TopSQP) uses information of the exact Hessian to define $\mathbf{B}_k$, while the IQP of the SQP+ implemented in this article uses a limited-memory BFGS approximation based on Byrd et al. (1996).

An important aspect for large-scale problems is to solve this sub-problem as fast as possible. In this context, the IQP of TopSQP is solved using fast commercial solvers for large-scale quadratic problems, such as Gurobi (Gurobi Optimization 2013) or CPLEX (IBM Corporation 2014).

Since SQP+ uses limited-memory BFGS approximation, a special purpose interior point method (Vanderbei and Shanno 1999) is implemented for solving the IQP sub-problem (Byrd et al. 1996).

2.3 Solving EQP sub-problem

The working set (Step 5 of Algorithm 1) contains all the indices of the constraints where their linearization in the sub-problem (IQP) are active at $\mathbf{d}_k^{iq}$, i.e,

$$\mathcal{W}_k^g = \{\ i\ \mid\ g_i(\mathbf{x}_k) + \nabla g_i(\mathbf{x}_k)^T \mathbf{d}_k^{iq} = 0,\ \ i = 1, \ldots, m\},$$
$$\mathcal{W}_k^u = \{\ i\ \mid\ (\mathbf{d}_k^{iq})_i - \tilde{u}_i = 0,\ \ i = 1, \ldots, n\},$$
$$\mathcal{W}_k^l = \{\ i\ \mid\ (\mathbf{d}_k^{iq})_i - \tilde{l}_i = 0,\ \ i = 1, \ldots, n\}.$$

For those active constraints, the following equality constrained sub-problem (EQP) is defined. The explanation of

its final formulation can be found in Byrd et al. (2004).

$$\begin{aligned} \underset{\mathbf{d}\in\mathbb{R}^n}{\text{minimize}} \;& (\nabla f(\mathbf{x}_k) + \mathbf{H}_k \mathbf{d}_k^{iq})^T \mathbf{d} + \tfrac{1}{2}\mathbf{d}^T \mathbf{H}_k \mathbf{d} \\ \text{subject to} \;& J_{\mathcal{W}_k^g}(\mathbf{x}_k)\mathbf{d} = \mathbf{0}, \\ & d_i = 0 \qquad\qquad i \in \{\mathcal{W}_k^u \cup \mathcal{W}_k^l\}. \end{aligned} \tag{EQP}$$

The matrix $\mathbf{H}_k$ refers to the Hessian of the Lagrangian function i.e. $\mathbf{H}_k = \nabla^2 \mathcal{L}(\mathbf{x}_k, \boldsymbol{\lambda}_k^{iq})$ and $J_{\mathcal{W}_k^g}(\mathbf{x}_k)$ represents the matrix whose rows are the active constraint gradients (i.e. the $\mathcal{W}_k^g$ rows of the Jacobian). Applying Newton's method to the first-order optimality conditions of (EQP), this optimization problem is equivalent to solve the KKT system

$$\begin{pmatrix} \mathbf{H}_k & J_{\mathcal{W}_k^g}(\mathbf{x}_k)^T & \mathbf{I}_{\mathcal{W}_k^u} & -\mathbf{I}_{\mathcal{W}_k^l} \\ J_{\mathcal{W}_k^g}(\mathbf{x}_k) & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{I}_{\mathcal{W}_k^u}^T & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ -\mathbf{I}_{\mathcal{W}_k^l}^T & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{d}_k^{eq} \\ \boldsymbol{\lambda}_k^{eq} \\ \boldsymbol{\xi}_k^{eq} \\ \boldsymbol{\eta}_k^{eq} \end{pmatrix} = - \begin{pmatrix} \nabla f(\mathbf{x}_k) + \mathbf{H}_k \mathbf{d}_k^{iq} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix} \tag{10}$$

where $\mathbf{I}_{\mathcal{W}_k^u} \in \mathbb{R}^{n\times|\mathcal{W}_k^u|}$ and $\mathbf{I}_{\mathcal{W}_k^l} \in \mathbb{R}^{n\times|\mathcal{W}_k^l|}$ are pseudo-identity matrices. The Lagrangian multipliers $(\boldsymbol{\lambda}_k^{eq}, \boldsymbol{\xi}_k^{eq}, \boldsymbol{\eta}_k^{eq})$ refer to the active constraints. The size of the system (10) can be easily reduced since

$$\begin{aligned} \mathbf{I}_{\mathcal{W}_k^u}^T \mathbf{d}_k^{eq} &= \mathbf{0}, \\ \mathbf{I}_{\mathcal{W}_k^l}^T \mathbf{d}_k^{eq} &= \mathbf{0}. \end{aligned}$$

Therefore, only a system taking into account the nonzero direction $((\mathbf{d}_k^{eq})_i \neq 0)$ need to be solved.

$$\begin{pmatrix} \mathbf{H}_{k,\mathcal{W}_{k,c}^b} & J_{\mathcal{W}_k^g,\mathcal{W}_{k,c}^b}(\mathbf{x}_k)^T \\ J_{\mathcal{W}_k^g,\mathcal{W}_{k,c}^b}(\mathbf{x}_k) & \mathbf{0} \end{pmatrix} \begin{pmatrix} \tilde{\mathbf{d}}_k^{eq} \\ \boldsymbol{\lambda}_k^{eq} \end{pmatrix} = - \begin{pmatrix} (\nabla f(\mathbf{x}_k) + \mathbf{H}_k \mathbf{d}_k^{iq})_{\mathcal{W}_{k,c}^b} \\ \mathbf{0} \end{pmatrix}. \tag{11}$$

Here, $\mathcal{W}_{k,c}^b = \{1,\ldots,n\} \setminus \{\mathcal{W}_k^u \cup \mathcal{W}_k^l\}$ is the complementarity set of the active bounds (upper and lower), $\mathbf{H}_{k,\mathcal{W}_{k,c}^b}$ refers to the $\mathcal{W}_{k,c}^b$ columns and rows of $\mathbf{H}_k$, and $J_{\mathcal{W}_k^g,\mathcal{W}_{k,c}^b}(\mathbf{x}_k)$ refers to the $\mathcal{W}_k^g$ rows and $\mathcal{W}_{k,c}^b$ columns of the Jacobian. In this case, the term $\tilde{\mathbf{d}}_k^{eq}$ represents the $\mathcal{W}_{k,c}^b$ rows of $\mathbf{d}_k^{eq}$. The Lagrangian multipliers of the active bounds are obtained afterwards[1] by

$$\begin{aligned} \mathbf{z}_k^{eq} &= -\nabla f(\mathbf{x}_k) - \mathbf{H}_k \mathbf{d}_k^{iq} - \mathbf{H}_k \mathbf{d}_k^{eq} - J_{\mathcal{W}_k^g}(\mathbf{x}_k)^T \boldsymbol{\lambda}_k^{eq}, \\ \boldsymbol{\xi}_k^{eq} &= \max(0, \mathbf{z}_k), \\ \boldsymbol{\eta}_k^{eq} &= -\min(0, \mathbf{z}_k). \end{aligned}$$

[1] Note that $\mathcal{W}_k^u$ and $\mathcal{W}_k^l$ are disjoint sets by definition.

The computation of the EQP search direction is, generally, much faster than solving IQP. The EQP sub-problem not only helps to produce a more accurate search direction, but also reduces the number of IQP phases, i.e. the number of iterations is decreased (see Section 7). A direct consequence is a reduction of the total computational time.

2.3.1 Existence of solutions of the equality quadratic constraint problem

The equality constrained quadratic problem (EQP) is equivalent to solving the system of (11). The matrix of this system is commonly defined like

$$\mathbf{K} = \begin{bmatrix} \mathbf{H} & \mathbf{A}^T \\ \mathbf{A} & \mathbf{0} \end{bmatrix},$$

where $\mathbf{H} \in \mathbb{R}^{n\times n}$ is symmetric and $\mathbf{A} \in \mathbb{R}^{m\times n}$ is the Jacobian of the linearized active constraints. It is assumed that $\mathbf{A}$ has full row rank. The matrix $\mathbf{K}$ is called a Karush-Kuhn-Tucker matrix. There are conditions under which the system has solution.

Theorem 1 (from Nocedal and Wright (1999)) *Let $\boldsymbol{A}$ have a full row rank, and assume that the reduced-Hessian matrix $\boldsymbol{Z}^T\boldsymbol{H}\boldsymbol{Z}$ is positive definite. Then the KKT matrix $\boldsymbol{K}$ is nonsingular, and hence there is a unique vector that satisfied the linear system and is the unique global solution.*

Where $\boldsymbol{Z} \in \mathbb{R}^{n\times(n-m)}$ is a matrix whose columns are the basis of the null-space of $\boldsymbol{A}$.

Thus, in order to obtain a solution of the EQP sub-problem, the reduced-Hessian must be positive definite, and the matrix $\mathbf{A}$ must have full rank.

Definition 1 (from Gould (1985)) The **inertia** of a symmetric matrix $\mathbf{W}$ is the triple (i_+, i_-, i_0), where i_0, i_+ and i_- are the number of zero, positive and negative eigenvalues of $\mathbf{W}$, respectively.

Theorem 2 ***Gould Theorem*** (Higham and Cheng 1998). *Suppose that $\boldsymbol{A}$ has full rank m. The condition $\boldsymbol{p}^T\boldsymbol{H}\boldsymbol{p} > 0$, $\forall \boldsymbol{p} \neq \boldsymbol{0}$, such that $\boldsymbol{A}^T\boldsymbol{p} = \boldsymbol{0}$ holds if and only if*

$$inertia(\boldsymbol{K}) = (n, m, 0).$$

In addition,

$$inertia(\boldsymbol{K}) = inertia(\boldsymbol{Z}^T\boldsymbol{H}\boldsymbol{Z}) + (m, m, 0).$$

Therefore, if $\boldsymbol{Z}^T\boldsymbol{H}\boldsymbol{Z}$ is positive definite, $inertia(\boldsymbol{K}) = (n, m, 0)$.

An option to compute the inertia of a KKT matrix is by the LDL factorization, see e.g. Forsgren and Murray (1997)

and Nocedal and Wright (1999). If the inertia is not correct, it is necessary to modify the KKT matrix to ensure the existence of solution. There are many different alternatives to modify the nonconvex sub-problem into a local convex approximation, such as using different inertia correction based on LDL factorizations, see e.g. Forsgren and Murray (1997), Forsgren and Gill (1998), and Forsgren (2002), or using other techniques such as in Higham and Cheng (1998) and Wächter and Biegler (2006). In addition, it is possible to apply different convexification approaches proposed in Gill and Robinson (2013) and Gill and Wong (2014).

When the computation of the reduced-Hessian is computationally cheap, the system can be easily modified to guarantee a positive definite reduced-Hessian, see Nocedal and Wright (1999). For instance, $\mathbf{H}_z = \mathbf{Z}_k^T \mathbf{H}_k \mathbf{Z}_k$ is perturbed using its eigenvalues, so that $\hat{\mathbf{H}}_z = \mathbf{H}_z + \gamma \mathbf{I} \succ 0$, with $\gamma = \min(|\lambda_{eig}(\mathbf{H}_z)|) + \epsilon$, where λ_{eig} refers the the eigenvalues of the matrix. This perturbation is then used for the KKT matrix,

$$\begin{bmatrix} \mathbf{H} + \gamma \mathbf{I} & \mathbf{A}^T \\ \mathbf{A} & \mathbf{0} \end{bmatrix}.$$

The EQP sub-problem in Morales et al. (2010) is solved only in those cases where the inertia of the system is correct. However, both the implemented SQP+ and TopSQP modifies the KKT matrix to guarantee a correct inertia. If the same approach as in Morales et al. (2010) is used for this particular problem, the solvers would essentially be a classical SQP with only an IQP step. The indefiniteness of the Hessian of the compliance (see Section 3) usually produces an incorrect inertia.

2.4 Acceptance/rejection of the step

Once the IQP and EQP search directions are computed, it is necessary to verify whether these estimates improve the iterate $\mathbf{x}_k$. First of all, a contraction parameter β (Step 7) secures that the linearization of all the constraints (active and inactive) are satisfied at the point with maximum search direction $\mathbf{d} = \mathbf{d}_k^{iq} + \beta \mathbf{d}_k^{eq}$. In other words, the constraints of the IQP phase must remain feasible.

The largest value of $\beta \in (0, 1]$ is computed such that

$$\begin{aligned} &g_i(\mathbf{x}_k) + \nabla g_i(\mathbf{x}_k)^T \mathbf{d} \leq 0 && i \in \mathcal{W}_{k,c}^g, \\ &d_i \leq \tilde{u}_i && i \in \mathcal{W}_{k,c}^u, \\ &\tilde{l}_i \leq d_i && i \in \mathcal{W}_{k,c}^l. \end{aligned}$$

Here,

$$\begin{aligned} \mathcal{W}_{k,c}^g &= \{1, \ldots, m\} \setminus \mathcal{W}_k^g, \\ \mathcal{W}_{k,c}^u &= \{1, \ldots, n\} \setminus \mathcal{W}_k^u, \\ \mathcal{W}_{k,c}^l &= \{1, \ldots, n\} \setminus \mathcal{W}_k^l, \end{aligned}$$

are the complementarity working set. Once the contraction parameter is determined, a line search estimates the step length α for the new step direction (Step 9 and 12 in Algorithm 1). The TopSQP line search is implemented following Morales et al. (2010). This line search procedure is slightly modified compared to the procedure used for the theoretical results. The acceptance criterion is based on the merit function (12) and the model reduction from $\mathbf{x}_k$ to $\mathbf{x}_k + \mathbf{d}$ (13) of the original problem (NLP). These functions are defined as

$$\phi_\pi(\mathbf{x}) = f(\mathbf{x}) + \pi(||\mathbf{g}(\mathbf{x})^+||_1 + ||\mathbf{g}_l(\mathbf{x})^-||_1 + ||\mathbf{g}_u(\mathbf{x})^-||_1) \tag{12}$$

$$\begin{aligned} qred_\pi(\mathbf{d}) = &-(\nabla f(\mathbf{x}_k)^T \mathbf{d} + \frac{1}{2} \mathbf{d}^T \mathbf{B}_k \mathbf{d}) + \pi(||\mathbf{g}(\mathbf{x}_k)^+||_1 \\ &+ ||\mathbf{g}_l(\mathbf{x}_k)^-||_1 + ||\mathbf{g}_u(\mathbf{x}_k)^-||_1). \end{aligned} \tag{13}$$

If the sufficient decrease condition (14) is satisfied, then the iterate $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{d}_k^{iq} + \beta \mathbf{d}_k^{eq}$ is accepted.

$$\phi_\pi(\mathbf{x}_k + \mathbf{d}_k^{iq} + \beta \mathbf{d}_k^{eq}) \leq \phi_\pi(\mathbf{x}_k) - \sigma qred_\pi(\mathbf{d}_k^{iq}). \tag{14}$$

Otherwise, $\alpha \in \{1, \kappa, \kappa^2, \ldots\}$ is found such that condition (15) is satisfied. In this case, the new iterate is defined as $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha \mathbf{d}_k^{iq}$.

$$\phi_\pi(\mathbf{x}_k + \alpha \mathbf{d}_k^{iq}) \leq \phi_\pi(\mathbf{x}_k) - \sigma \alpha qred_\pi(\mathbf{d}_k^{iq}). \tag{15}$$

There are several techniques to update the penalty parameter π in order to improve the convergence rate, for instance, the strategy explained in e.g. Wang and Pu (2013) and Curtis and Nocedal (2008). In practice, due to the feasibility of the sub-problems (see Section 3), the term affected by π is always close to 0. The penalty parameter π is updated very simple by just using the Lagrangian multipliers, i.e. $\pi = ||\lambda||_\infty$. Finally, the parameters $\sigma = 10^{-4}$ and $\kappa = 0.5$ are taken from Morales et al. (2010).

2.5 Updating the Lagrangian multipliers

The SQP+ algorithm in Morales et al. (2010) updates the estimates of the Lagrangian multipliers depending on the final step direction. The updating scheme of the SQP+ is

$$\boldsymbol{\lambda}_{k+1} = \begin{cases} \max(\mathbf{0}, \boldsymbol{\lambda}_k^{eq}) & \text{if } \mathbf{d}_k = \mathbf{d}_k^{iq} + \beta \mathbf{d}_k^{eq} \\ (1 - \alpha)\boldsymbol{\lambda}_k + \alpha \boldsymbol{\lambda}_k^{iq} & \text{if } \mathbf{d}_k = \alpha \mathbf{d}_k^{iq}. \end{cases}$$

Although Morales et al. (2010) suggest that both the equality and the inequality Lagrangian multipliers can be good candidates, the former is considered due to the use of BFGS approximations in the IQP phase and the exact Hessian in the EQP phase. Nevertheless, a preliminary study was performed to investigate how the election of the Lagrangian multipliers was affecting the convergence. The numerical experiments show that for the proposed implementation, the

inequality Lagrangian multipliers are more accurate. This might be due to the use of an approximate Hessian in the EQP phase. In addition, the working set of active constraints is numerically approximated (see Section 7), and the EQP might be defined with constraints that are inactive. Thus, the updating scheme is as follows,

$$\begin{aligned}\boldsymbol{\lambda}_{k+1} &= \alpha\boldsymbol{\lambda}_k^{iq} + (1-\alpha)\boldsymbol{\lambda}_k,\\ \boldsymbol{\xi}_{k+1} &= \alpha\boldsymbol{\xi}_k^{iq} + (1-\alpha)\boldsymbol{\xi}_k,\\ \boldsymbol{\eta}_{k+1} &= \alpha\boldsymbol{\eta}_k^{iq} + (1-\alpha)\boldsymbol{\eta}_k.\end{aligned}$$

The same conclusion was obtained for the general SQP+ implemented for the numerical experiments.

2.6 Convergence properties

In Morales et al. (2010) the global and local convergence properties of SQP+ are explained in detail. The IQP phase gives the global convergence while the EQP phase helps to produce fast local convergence, when the active set is correct (Morales et al. 2010). For the theoretical proof, some assumptions are established. In addition, the quadratic convergence proof relies on the second-order sufficient conditions[2] at $\mathbf{x}_k$ sufficiently close to the optimal point. A full step ($\mathbf{d}_k = \mathbf{d}_k^{iq} + \mathbf{d}_k^{eq}$) is also assumed at those points (Morales et al. 2010). Even, if it is not the goal of this article to demonstrate the theoretical convergence of the solver, it is important to point out that some of the assumptions made in Morales et al. (2010) are easily proven for our specific problem, such as the convexity of the feasible set, the feasibility of the IQP sub-problem and the Lipschitz continuous property of the objective and constraint functions. On the other hand, there are several assumptions that cannot be proven for general topology optimization problems, such as the strict complementarity assumption. In general, these assumptions are quite strict and they exist to prove general global and local convergence.

In practice, TopSQP may loose the theoretical quadratic convergence property. Due to numerical tolerances, at iterates close to the optimal solution, the optimization method does not take the full step, i.e. $\beta < 1$. The strict complementarity is not satisfied, and at some of these iterations, the second-order sufficient conditions are not satisfied. Thus, Theorem 4.3 in Morales et al. (2010) cannot be applied. Nevertheless, the numerical experiments in Section 8 show that the proposed implementation enjoys great robustness.

[2](from Nocedal and Wright (1999)) For a given feasible point $\bar{\mathbf{x}}$, there are some Lagrangian multipliers $(\bar{\boldsymbol{\lambda}}, \bar{\boldsymbol{\xi}}, \bar{\boldsymbol{\eta}})$ such that the KKT conditions are satisfied. Suppose that $\mathbf{p}^T\mathbf{H}\mathbf{p} > 0$,such that $\mathbf{A}^T\mathbf{p} = \mathbf{0}$, with $\mathbf{A}$ the Jacobian of the active constraints. Then, $\bar{\mathbf{x}}$ is a strict local solution.

3 Problem formulation

The minimum compliance problem is considered one of the most typical structural topology optimization problems. The classical formulation consists of maximizing the stiffness of the structure (minimizing compliance) subject to a volume constraint, see more details in e.g. Bendsøe and Sigmund (2003). This article considers the nested approach, where the displacements (state variables, $\mathbf{u}$) depend on the design variables ($\mathbf{t}$), related with the linear elastic equilibrium equations in their discretized form

$$\begin{aligned}&\mathbf{K}(\mathbf{t})\mathbf{u} = \mathbf{f},\\ &\mathbf{u}(\mathbf{t}) = \mathbf{K}^{-1}(\mathbf{t})\mathbf{f}.\end{aligned}$$

Here $\mathbf{t} \in \mathbb{R}^n$ is the density variable, n is the number of elements, $\mathbf{K} : \mathbb{R}^n \to \mathbb{R}^{d\times d}$ is the stiffness matrix, with d the number of degrees of freedom, and $\mathbf{f} \in \mathbb{R}^d$ the static design-independent force vector.

In particular, the density-based approach is used to penalize intermediate densities to produce an almost solid-and-void design. More specifically, the Solid Isotropic of Material Penalization (SIMP) approach is chosen (see e.g. Bendsøe (1989), Rozvany et al. (1992), and Zhou and Rozvany (1991)). For this interpolation, the stiffness matrix is defined as

$$\mathbf{K}(\mathbf{t}) = \sum_{e=1}^{n} E(t_e)\mathbf{K}_e,$$

with

$$E(t_e) = E_v + (E_1 - E_v)\tilde{t}_e^p,$$

where the SIMP penalty parameter is $p \geq 1$. Here, $E_v > 0$ and $E_1 \gg E_v$ are the "void" and solid Young's modulus, respectively, and $\mathbf{K}_e$ the element stiffness matrix. The stiffness matrix is assumed to be positive definite for all design vectors satisfying the bound constraints to avoid singularity. Finally, the variable $\tilde{t}_e$ refers to the design variable with a density filter (Bourdin 2001), and (Sigmund and Petersson 1998), defined analogous to Rojas-Labanda and Stolpe (2015).

The minimum compliance problem in its discrete version is

$$\begin{aligned}\underset{\mathbf{t}\in\mathbb{R}^n}{\text{minimize}}\ & \mathbf{u}(\mathbf{t})^T\mathbf{K}(\mathbf{t})\mathbf{u}(\mathbf{t})\\ \text{subject to}\ & \mathbf{a}^T\mathbf{t} \leq V,\\ & \mathbf{0} \leq \mathbf{t} \leq \mathbf{1}.\end{aligned} \qquad (P^c)$$

Here $\mathbf{a} = (a_1, \ldots, a_n)^T \in \mathbb{R}^n$ with $a_i > 0 \quad i = 1, \ldots, n$ the relative element volume, and $V > 0$ the total volume fraction. For simplicity, $a_i = a_j \quad \forall i, j$.

The nonlinear optimization problem (P^c) contains only one linear inequality constraint and bound constraints. Denote the feasible set of (P^c) by

$$\Omega = \{t_i \in [0,1] \quad i = 1, \ldots, n, \sum_{i=1}^{n} a_i t_i \leq V\}. \qquad (16)$$

The set Ω in (16) is convex, nonempty under natural assumptions, closed, bounded and thus compact, (Boyd and Vandenberghe 2010).

Both Ω and the constraint functions are convex. However, the optimization problem is, in general, nonconvex (Boyd and Vandenberghe 2010) since the Hessian of the Lagrangian function $\nabla^2\mathcal{L}(\mathbf{t},\boldsymbol{\lambda}) = \nabla^2 f(\mathbf{x})$ is not positive semi-definite (cf. below). The feasible set is nonempty, i.e. there is, at least, one local solution for (NLP). Certain CQs hold at every point due to the linearity of the constraints. The authors emphasize the importance of the CQ because, in general, the numerical optimization theory assumes that they are satisfied (see e.g. Nocedal and Wright (1999) and Luenberger and Ye (2008)).

4 Approximation of the hessian of the lagrangian

The Hessian is defined by using direct sensitivity analysis on the objective function,

$$\nabla^2\mathcal{L}(\mathbf{t},\boldsymbol{\lambda}) = 2\mathbf{F}(\mathbf{t})^T\mathbf{K}^{-1}(\mathbf{t})\mathbf{F}(\mathbf{t}) - \mathbf{Q}(\mathbf{t})$$

with

$$\mathbf{Q}(\mathbf{t}) = \operatorname{diag}(\mathbf{u}^T(\mathbf{t})\frac{\partial^2\mathbf{K}_i(t_i)}{\partial t_i^2}\mathbf{u}(\mathbf{t})) : \mathbb{R}^n \longrightarrow \mathbb{R}^{n\times n}$$

$$\mathbf{F}(\mathbf{t}) = \left(\frac{\partial\mathbf{K}_1(t_1)}{\partial t_1}\mathbf{u}(\mathbf{t}) \cdots \frac{\partial\mathbf{K}_n(t_n)}{\partial t_n}\mathbf{u}(\mathbf{t})\right) : \mathbb{R}^n \longrightarrow \mathbb{R}^{d\times n}.$$

For $p = 1$ (SIMP penalization parameter), the term $\mathbf{Q}(\mathbf{t})$ is zero, and the problem is convex with

$$\nabla^2\mathcal{L}(\mathbf{t},\lambda) = 2\mathbf{F}(\mathbf{t})^T\mathbf{K}^{-1}(\mathbf{t})\mathbf{F}(\mathbf{t}) \succeq 0.$$

The problem generally becomes nonconvex for $p > 1$.

The IQP phase requires a positive definite approximation of the Hessian, $\mathbf{B}_k$ (Step 3 in Algorithm 1). Instead of using a BFGS approximation as in Morales et al. (2010), second-order information is used as much as possible. A convex (positive semi-definite) approximation could be for instance,

$$\hat{\mathbf{H}}_1 = 2\mathbf{F}(\mathbf{t})^T\mathbf{K}^{-1}(\mathbf{t})\mathbf{F}(\mathbf{t}). \tag{17}$$

From a theoretical point of view, the approximate matrix $\mathbf{B}_k$ must be positive definite, i.e. $\mathbf{B}_k = \hat{\mathbf{H}}_1 + \epsilon\mathbf{I} \succ 0$. In practice, $\mathbf{B}_k = \hat{\mathbf{H}}_1 \succeq 0$ is used, and thus, some theoretical properties might be lost. In the numerical experiments, the approximate Hessian $\hat{\mathbf{H}}_1$ is strictly positive definite, though.

There are other alternatives to modify the Hessian so that it becomes positive definite, such as (18) where an extra term is added, or (19) where the nonconvex part is contracted by a factor γ_2.

$$\hat{\mathbf{H}}_2 = 2\mathbf{F}(\mathbf{t})^T\mathbf{K}^{-1}(\mathbf{t})\mathbf{F}(\mathbf{t}) - \mathbf{Q}(\mathbf{t}) + \gamma_1\mathbf{I}, \tag{18}$$

$$\hat{\mathbf{H}}_3 = 2\mathbf{F}(\mathbf{t})^T\mathbf{K}^{-1}(\mathbf{t})\mathbf{F}(\mathbf{t}) - \gamma_2\mathbf{Q}(\mathbf{t}). \tag{19}$$

There are some other alternatives to $\mathbf{B}_k$, for instance, the identity matrix, or the positive diagonal terms of the exact Hessian, as it is suggested in Fleury (1989b). Nevertheless, we would like to take advantage of the structure of the exact Hessian. The positive semi-definite $\hat{\mathbf{H}}_1$ (17) matrix is chosen to be the $\mathbf{B}_k$ of IQP step, since it is the fastest approximation (there is no need to estimate γ_1 or γ_2).

In the EQP step, either the exact Hessian with an inertia correction method is used or the KKT matrix is perturbed so that the reduced-Hessian becomes positive definite.

The minimum compliance problem (P^c) has only one linear inequality constraint, therefore, a basis ($\mathbf{Z}$) of the null-space of the gradient of the active constraints is very easy to compute. A perturbation of the KKT matrix is easily obtained by enforcing a positive definite reduced-Hessian introduced in Section 2.3.1. However, the reduced-Hessian, $\mathbf{Z}^T\mathbf{H}\mathbf{Z}$, becomes dense and the computation of γ will be expensive.

On the other hand, it is also possible to directly use the same positive definite approximate Hessian as in the IQP. This positive definite approximation guarantees the correctness of the inertia of the system without examines it. However, once the active constraint set is identified, both the EQP and the IQP will theoretically be identical (Morales et al. 2010). In practice, the set of active constraints of the considered problem is identified at the very end of the optimization process since the density variables are constantly moving. Moreover, the numerical errors in both, the tolerance of the IQP and EQP steps, and the selection of the active set (see Section 7), produce that the IQP and EQP steps at points sufficiently close to the optimal are different.

5 Alternatives to the primal IQP formulation

The IQP sub-problem of (P^c) is formulated using an approximate Hessian $\mathbf{B}_k$ defined in (17) as

$$\begin{aligned} \underset{\mathbf{d}\in\mathbb{R}^n}{\text{minimize}} \quad & \hat{f}(\mathbf{d}) = -(\mathbf{F}_k^T\mathbf{u}_k)^T\mathbf{d} + \mathbf{d}^T\mathbf{F}_k^T\mathbf{K}_k^{-1}\mathbf{F}_k\mathbf{d} \\ \text{subject to} \quad & \mathbf{a}^T\mathbf{t}_k - V + \mathbf{a}^T\mathbf{d} \le 0 \\ & -\mathbf{t}_k \le \mathbf{d} \le \mathbf{1} - \mathbf{t}_k. \end{aligned} \tag{IQP_p}$$

For simplicity, any function or matrix with the form $A(\mathbf{t}_k)$ is represented by A_k.

The feasible set of (IQP_p) is convex (hyperplanes), nonempty, closed, and bounded. By construction the (IQP_p) is convex. Moreover some CQs hold. Thus, this sub-problem has an optimal solution and any local solution that satisfies the KKT condition is also a global minimizer.

5.1 Reformulation of the primal IQP formulation

TopSQP spends most of the computational time in the IQP phase. In addition, (IQP_p) is extremely expensive since the

inverse of the stiffness matrix is involved, and the matrix $\mathbf{B}_k$ is dense.

However, it is possible to reformulate the inequality sub-problem such that the computation and the storage of the approximate Hessian are no longer required. First of all, a Cholesky factorization is used for the stiffness matrix. Then, a new variable $\tilde{\mathbf{z}} = (\mathbf{R}_k^T)^{-1}\mathbf{F}_k\mathbf{d} \in \mathbb{R}^d$ is included to rename some terms of the problem so that any inverse matrix is removed from the objective function of (IQP_p).

$$\hat{f}(\mathbf{d}) = -(\mathbf{F}_k^T\mathbf{u}_k)^T\mathbf{d} + \mathbf{d}^T\mathbf{F}_k^T(\mathbf{R}_k^T\mathbf{R}_k)^{-1}\mathbf{F}_k\mathbf{d},$$
$$\hat{f}(\mathbf{d}, \tilde{\mathbf{z}}) = -(\mathbf{F}_k^T\mathbf{u}_k)^T\mathbf{d} + \tilde{\mathbf{z}}^T\tilde{\mathbf{z}}.$$

The introduction of the new variable $\tilde{\mathbf{z}}$ leads to an enlargement of the number of constraints. The alternative IQP formulation is

$$\begin{aligned} \underset{\mathbf{d}\in\mathbb{R}^n, \tilde{\mathbf{z}}\in\mathbb{R}^d}{\text{minimize}} \quad & -(\mathbf{F}_k^T\mathbf{u}_k)^T\mathbf{d} + \tilde{\mathbf{z}}^T\tilde{\mathbf{z}} \\ \text{subject to} \quad & \mathbf{R}_k^T\tilde{\mathbf{z}} - \mathbf{F}_k\mathbf{d} = \mathbf{0}, \\ & \mathbf{A}_k\mathbf{d} \le \mathbf{b}_k, \end{aligned} \qquad (IQP_{p-2})$$

where the linear inequality constraints are condensed into a system $\mathbf{A}_k\mathbf{d} \le \mathbf{b}_k$, to simplify the notation. Here, $m = 2n + 1$, and

$$\mathbf{A}_k = \begin{bmatrix} \mathbf{a}^T \\ \mathbf{I} \\ -\mathbf{I} \end{bmatrix} \in \mathbb{R}^{m\times n},$$
$$\mathbf{b}_k = \begin{bmatrix} -(\mathbf{a}^T\mathbf{t}_k - V) \\ \mathbf{1} - \mathbf{t}_k \\ \mathbf{t}_k \end{bmatrix} \in \mathbb{R}^m.$$

Using this new formulation, the number of variables and linear constraints are increased. In contrast, the computational time can, due to sparsity, significantly be reduced.

5.2 Dual problem of the IQP formulation

Using Lagrangian duality theory (see e.g. Boyd and Vandenberghe (2010)), an optimization problem can be reformulated using the *dual variables* $(\boldsymbol{\lambda}, \boldsymbol{\xi}, \boldsymbol{\eta})$. In some cases this new *dual problem* is much easier to solve and computationally less expensive than the primal problem. Thus, the problem (IQP_p) can also be formulated in its dual problem.

A new variable $\mathbf{z} \in \mathbb{R}^d$ is introduced to rename some terms of the objective function. Let

$$\mathbf{z} = \mathbf{F}_k\mathbf{d}, \qquad (20)$$

then, the (IQP_p) problem is equivalent to

$$\begin{aligned} \underset{\mathbf{d}\in\mathbb{R}^n, \mathbf{z}\in\mathbb{R}^d}{\text{minimize}} \quad & -(\mathbf{F}_k^T\mathbf{u}_k)^T\mathbf{d} + \mathbf{z}^T\mathbf{K}_k^{-1}\mathbf{z} \\ \text{subject to} \quad & \mathbf{A}_k\mathbf{d} \le \mathbf{b}_k, \\ & \mathbf{z} = \mathbf{F}_k\mathbf{d}. \end{aligned} \qquad (21)$$

The Lagrangian function of the IQP sub-problem (21) is described in terms of the primal variables $\mathbf{d}$ and $\mathbf{z}$, and the dual variables $\boldsymbol{\nu} = (\boldsymbol{\lambda}, \boldsymbol{\xi}, \boldsymbol{\eta}) \in \mathbb{R}^m$ (for the inequality and bound constraints) and $\boldsymbol{\theta} \in \mathbb{R}^d$ (for the new equality constraints).

$$\begin{aligned} \mathcal{L}_p(\mathbf{d}, \mathbf{z}, \boldsymbol{\nu}, \boldsymbol{\theta}) = & -(\mathbf{F}_k^T\mathbf{u}_k)^T\mathbf{d} + \mathbf{z}^T\mathbf{K}_k^{-1}\mathbf{z} \\ & + \boldsymbol{\nu}^T(\mathbf{A}_k\mathbf{d} - \mathbf{b}_k) + \boldsymbol{\theta}^T(\mathbf{z} - \mathbf{F}_k\mathbf{d}). \end{aligned}$$

The dual problem consists of maximizing

$$\varphi(\boldsymbol{\nu}, \boldsymbol{\theta}) = \inf_{\mathbf{d},\mathbf{z}} \mathcal{L}_p(\mathbf{d}, \mathbf{z}, \boldsymbol{\nu}, \boldsymbol{\theta})$$

respect to the dual variables ν, θ (Boyd and Vandenberghe 2010). The formulation of the dual problem is obtained by satisfying the optimality conditions of (21),

$$\nabla_{\mathbf{d}}\mathcal{L}_p(\bar{\mathbf{d}}, \bar{\mathbf{z}}, \bar{\boldsymbol{\nu}}, \bar{\boldsymbol{\theta}}) = -\mathbf{F}_k^T\mathbf{u}_k + \mathbf{A}_k^T\bar{\boldsymbol{\nu}} - \mathbf{F}_k^T\bar{\boldsymbol{\theta}} = \mathbf{0},$$
$$\nabla_{\mathbf{z}}\mathcal{L}_p(\bar{\mathbf{d}}, \bar{\mathbf{z}}, \bar{\boldsymbol{\nu}}, \bar{\boldsymbol{\theta}}) = 2\mathbf{K}_k^{-1}\bar{\mathbf{z}} + \bar{\boldsymbol{\theta}} = \mathbf{0}.$$

Then, the solution of the dual problem must satisfy:

$$\bar{\mathbf{z}} = -\frac{1}{2}\mathbf{K}_k\bar{\boldsymbol{\theta}}, \qquad (22)$$
$$-\mathbf{F}_k^T\mathbf{u}_k + \mathbf{A}_k^T\bar{\boldsymbol{\nu}} - \mathbf{F}_k^T\bar{\boldsymbol{\theta}} = \mathbf{0}. \qquad (23)$$

Based on the above equations, the primal Lagrangian function at $(\bar{\mathbf{d}}, \bar{\mathbf{z}}, \bar{\nu}, \bar{\theta})$ is

$$\begin{aligned} \mathcal{L}_p(\bar{\mathbf{d}}, \bar{\mathbf{z}}, \bar{\boldsymbol{\nu}}, \bar{\boldsymbol{\theta}}) = & -(\mathbf{F}_k^T\mathbf{u}_k)^T\bar{\mathbf{d}} + \bar{\mathbf{z}}^T\mathbf{K}_k^{-1}\bar{\mathbf{z}} + \bar{\boldsymbol{\nu}}^T(\mathbf{A}_k\bar{\mathbf{d}} - \mathbf{b}_k) \\ & + \bar{\boldsymbol{\theta}}^T(\bar{\mathbf{z}} - \mathbf{F}_k\bar{\mathbf{d}}) \\ = & (-\mathbf{F}_k^T\mathbf{u}_k + \mathbf{A}_k^T\bar{\boldsymbol{\nu}} - \mathbf{F}_k^T\bar{\boldsymbol{\theta}})^T\bar{\mathbf{d}} \\ & - \frac{1}{4}\bar{\boldsymbol{\theta}}^T\mathbf{K}_k\bar{\boldsymbol{\theta}} - \bar{\boldsymbol{\nu}}^T\mathbf{b}_k. \end{aligned} \qquad (24)$$

Thus, the dual IQP problem is defined by merging (23) and (24) resulting in the quadratic problem

$$\begin{aligned} \underset{\boldsymbol{\nu}\in\mathbb{R}^m, \boldsymbol{\theta}\in\mathbb{R}^d}{\text{maximize}} \quad & \varphi(\boldsymbol{\nu}, \boldsymbol{\theta}) = -\tfrac{1}{4}\boldsymbol{\theta}^T\mathbf{K}_k\boldsymbol{\theta} - \boldsymbol{\nu}^T\mathbf{b}_k \\ \text{subject to} \quad & \mathbf{A}_k^T\boldsymbol{\nu} - \mathbf{F}_k^T\boldsymbol{\theta} = \mathbf{F}_k^T\mathbf{u}_k, \\ & \nu \ge \mathbf{0}, \end{aligned}$$

which is equivalent to

$$\begin{aligned} \underset{\nu,\theta}{\text{minimize}} \quad & \tfrac{1}{4}\theta^T\mathbf{K}_k\theta + \nu^T\mathbf{b}_k \\ \text{subject to} \quad & \mathbf{A}_k^T\nu - \mathbf{F}_k^T\theta = \mathbf{F}_k^T\mathbf{u}_k, \\ & \nu \ge \mathbf{0}. \end{aligned} \qquad (IQP_d)$$

The dual problem (IQP_d) is defined with a quadratic convex objective function, n linear equality constraints and $m = 2n + 1$ bound constraints. The strong duality property for convex problems is satisfied, see e.g. Boyd and Vandenberghe (2010). Thus, $\varphi(\bar{\boldsymbol{\nu}}, \bar{\boldsymbol{\theta}}) = \hat{f}(\bar{\mathbf{d}})$.

In order to recover the primal variables, the optimality conditions of the dual problem (IQP_d) are explicitly obtained. Given the dual Lagrangian function

$$\begin{aligned} \mathcal{L}_d(\boldsymbol{\nu}, \boldsymbol{\theta}, \boldsymbol{\chi}, \boldsymbol{\zeta}) = & \frac{1}{4}\boldsymbol{\theta}^T\mathbf{K}_k\boldsymbol{\theta} + \boldsymbol{\nu}^T\mathbf{b}_k \\ & + \boldsymbol{\chi}^T(-\mathbf{F}_k^T\mathbf{u}_k + \mathbf{A}_k^T\boldsymbol{\nu} - \mathbf{F}_k^T\boldsymbol{\theta}) - \boldsymbol{\zeta}^T\boldsymbol{\nu}, \end{aligned}$$

the optimality conditions are satisfied at $(\overline{\boldsymbol{\nu}}, \overline{\boldsymbol{\theta}}, \overline{\boldsymbol{\chi}}, \overline{\boldsymbol{\zeta}})$

$$\begin{aligned}\nabla_{\nu}\mathcal{L}_d(\overline{\boldsymbol{\nu}}, \overline{\boldsymbol{\theta}}, \overline{\boldsymbol{\chi}}, \overline{\boldsymbol{\zeta}}) &= \mathbf{b}_k + \mathbf{A}_k\overline{\boldsymbol{\chi}} - \overline{\boldsymbol{\zeta}} = \mathbf{0}\\ \nabla_{\theta}\mathcal{L}_d(\overline{\boldsymbol{\nu}}, \overline{\boldsymbol{\theta}}, \overline{\boldsymbol{\chi}}, \overline{\boldsymbol{\zeta}}) &= \frac{1}{2}\mathbf{K}_k\overline{\boldsymbol{\theta}} - \mathbf{F}_k\overline{\boldsymbol{\chi}} = \mathbf{0}.\end{aligned} \tag{25}$$

Here, $\boldsymbol{\chi} \in \mathbb{R}^n$ and $\boldsymbol{\zeta} \in \mathbb{R}^m$ are the Lagrangian multipliers of the equality and the bound constraints of (IQP_d), respectively.

From (25), $\overline{\boldsymbol{\theta}} = 2\mathbf{K}_k^{-1}\mathbf{F}_k\overline{\boldsymbol{\chi}}$ is obtained. In addition, the primal variable $\mathbf{z}$ is related with the dual variable $\boldsymbol{\theta}$ by (22). Thus, $\overline{\mathbf{z}} = -\mathbf{F}_k\overline{\boldsymbol{\chi}}$. Since $\mathbf{z}$ was previously defined as $\mathbf{z} = \mathbf{F}_k\mathbf{d}$ (20), the optimal primal variable $\overline{\mathbf{d}}$ is equivalent to the negative value of the optimal dual variable of (IQP_d), i.e. $\overline{\mathbf{d}}_k^{iq} = -\overline{\boldsymbol{\chi}}$.

The variable $\overline{\boldsymbol{\nu}}$ collects the inequality, the upper bound, and the lower bound Lagrangian multipliers, i.e. $\overline{\boldsymbol{\nu}} = (\overline{\boldsymbol{\lambda}_k^{iq}}, \overline{\boldsymbol{\xi}_k^{iq}}, \overline{\boldsymbol{\eta}_k^{iq}})$ with $\boldsymbol{\lambda}_k^{iq} \in \mathbb{R}$, and $\boldsymbol{\xi}_k^{iq}, \boldsymbol{\eta}_k^{iq} \in \mathbb{R}^n$.

The main advantage of solving the IQP sub-problem using the dual formulation is the elimination of the inverse of the stiffness matrix. This new formulation is expected to be faster than the alternative primal formulation since there are fewer number of variables and constraints. Therefore, it is chosen for the implementation of TopSQP.

6 Alternative to the EQP system

Analogous to the IQP, the KKT system for the minimum compliance problem is impractical since to the computation of the Hessian is needed. Throughout this section, and with the aim of simplifying the notation, the sub-indices referring to the working set or the complementary working set are omitted, see Section 2.3 for the sake of completeness.

The original EQP system for minimum compliance problem is

$$\begin{bmatrix}\hat{\mathbf{H}}_k & \mathbf{a}\\ \mathbf{a}^T & 0\end{bmatrix}\begin{bmatrix}\mathbf{d}_k^{eq}\\ \lambda_k^{eq}\end{bmatrix} = -\begin{bmatrix}-\mathbf{F}_k^T\mathbf{u}_k + \mathbf{H}_k\mathbf{d}_k^{iq}\\ 0\end{bmatrix} \qquad (EQP_0)$$

With $\mathbf{H}_k = \nabla^2\mathcal{L}(\mathbf{x}_k, \lambda_k^{iq})$, and $\hat{\mathbf{H}}_k$ an approximation of the Hessian such that the inertia of the system (EQP_0) is correct.

Let assume $\hat{\mathbf{H}}_k = 2\mathbf{F}_k^T\mathbf{K}_k^{-1}\mathbf{F}_k - \hat{\mathbf{Q}}_k$ for any $\hat{\mathbf{Q}}_k$ such that $\hat{\mathbf{H}}_k \succ 0$.

The first system of equations for the minimum compliance problem is

$$\begin{aligned}&\hat{\mathbf{H}}_k\mathbf{d}_k^{eq} + \mathbf{a}\lambda_k^{eq} = -(-\mathbf{F}_k^T\mathbf{u}_k + \mathbf{H}_k\mathbf{d}_k^{iq}),\\ &\qquad\Downarrow\\ &2\mathbf{F}_k^T\mathbf{K}_k^{-1}\mathbf{F}_k\mathbf{d}_k^{eq} - \hat{\mathbf{Q}}_k\mathbf{d}_k^{eq} + \mathbf{a}\lambda_k^{eq} = -(-\mathbf{F}_k^T\mathbf{u}_k + \mathbf{H}_k\mathbf{d}_k^{iq}).\end{aligned} \tag{26}$$

In order to reduce the computational cost caused for the dense Hessian, the system is expanded. A new variable $\mathbf{v}_k = 2\mathbf{K}_k^{-1}\mathbf{F}_k\mathbf{d}_k^{eq}$ is included to split (26) in two,

$$\begin{aligned}&\tfrac{1}{2}\mathbf{K}_k\mathbf{v}_k - \mathbf{F}_k\mathbf{d}_k^{eq} = \mathbf{0},\\ &\mathbf{F}_k^T\mathbf{v}_k - \hat{\mathbf{Q}}_k\mathbf{d}_k^{eq} + \mathbf{a}\lambda_k^{eq} = -(-\mathbf{F}_k^T\mathbf{u}_k + \mathbf{H}_k\mathbf{d}_k^{iq}).\end{aligned}$$

It enables to define an expanded EQP symmetric system

$$\begin{bmatrix}-\hat{\mathbf{Q}}_k & \mathbf{F}_k^T & \mathbf{a}\\ \mathbf{F}_k & -1/2\mathbf{K}_k & \mathbf{0}\\ \mathbf{a}^T & 0 & 0\end{bmatrix}\begin{bmatrix}\mathbf{d}_k^{eq}\\ \mathbf{v}_k\\ \lambda_k^{eq}\end{bmatrix} = -\begin{bmatrix}-\mathbf{F}_k^T\mathbf{u}_k + \mathbf{H}_k\mathbf{d}_k^{iq}\\ \mathbf{0}\\ 0\end{bmatrix}. \qquad (EQP_e)$$

Although the size of the system is increased from $|\mathcal{W}_{k,c}^b| + |\mathcal{W}_k^g|$ to $d + |\mathcal{W}_{k,c}^b| + |\mathcal{W}_k^g|$, it is much faster to solve than (EQP_0) since there are only sparse matrices.

The system can be solved using direct methods such as the null-space method (Nocedal and Wright 1999). The

Table 1 Study of the number of iterations required for TopSQP to converge using both EQP+IQP and using only the IQP phase on a test set of 10 small-size problems. The table contains the description of the problem (design domain, length ratios, discretization, and volume fraction) and the number of iterations required for both approaches

Domain	Length ratio	Discretization	Volume	TopIQP iterations	TopSQP iterations
Michell	1×1	20×20	0.1	63	36
Michell	1×1	40×40	0.3	85	74
Michell	2×1	40×20	0.1	256	151
Michell	2×1	80×40	0.5	37	30
Michell	3×1	60×20	0.4	137	114
MBB	1×2	40×80	0.3	88	59
MBB	1×4	40×160	0.5	124	113
MBB	2×1	80×40	0.2	169	141
Cantilever	2×1	120×60	0.5	81	78
Cantilever	4×1	80×20	0.2	131	92

Table 2 Parameter settings for TopSQP. Here, **e** refers to a vector of all ones. The table contains the name of the parameter, a brief description and the value

Parameter	Description	Value
$\mathbf{t}_0$	Starting point	$V\mathbf{e}$
`stat tol`	Stationarity error ϵ_1 (see Section 2.1)	10^{-6}
`feas tol`	Feasibility error ϵ_2 (see Section 2.1)	10^{-8}
`comp tol`	Complementarity error ϵ_3 (see Section 2.1)	10^{-6}
`max iter`	Maximum number of iterations	1,000

computation of a matrix $\mathbf{Z}$ with columns are the basis of the null-space of the Jacobian is cheap, easy, and fast. However, as there is only one constraint, the time reduction is negligible. In addition, the computational cost of the EQP step is not significant for the overall algorithm due to the sparsity of the KKT matrix. Moreover, the density variables tend fast to the bounds, and most of the bound constraints will be active. This produces a meaningful reduction of the size of the system (EQP_e), see (11).

7 Implementation

The same approximate Hessian is used for both the IQP and the EQP phases (see (17)) since preliminary results show that the performance of $\hat{\mathbf{H}}_1$ in the EQP was very similar to $\hat{\mathbf{H}}_2$ and $\hat{\mathbf{H}}_3$. However, the computational time required for $\hat{\mathbf{H}}_2$ and $\hat{\mathbf{H}}_3$ is higher than $\hat{\mathbf{H}}_1$ due to the estimation of γ_1 and γ_2. In addition any inertia correction in the original system will increase the computational time of the algorithm considerably.

The implementation of the proposed TopSQP algorithm is written in MATLAB (The MathWorks 2014), and the IQP sub-problem is solved using Gurobi optimizer software version 5.6.3 (Gurobi Optimization 2013). The default method in Gurobi is used in which the QP problem is solved with a barrier algorithm. Although the Gurobi software is very efficient, the IQP phase makes the method expensive. The EQP is, thus, very important to reduce the number of IQP iterations.

The numerical experiments in Morales et al. (2010) show the benefits of the EQP in terms of number of iterations. Since the proposed IQP is defined with the same approximate Hessian as the EQP phase, a small preliminary study was performed to study the effects of the EQP step. Table 1 shows the number of iterations needed for TopSQP using only the IQP (namely TopIQP) and using both phases. Although, only ten small problems are considered, similar behaviour was observed for the whole test set of problems. The EQP reduces the total number of iterations. In addition, the cost of the EQP phase is relatively small compared to the IQP.

For the estimation of the working set, the linearized constraint $g(\mathbf{x})$ are considered active if

$$-\epsilon_4 < g(\mathbf{x}_k) + \nabla g(\mathbf{x}_k)^T \mathbf{d}_k^{iq} < \epsilon_4,$$

with $\epsilon_4 = 10^{-4}$. In the same way, the line search is, in practice, more flexible. First of all, a smaller reduction in the merit function than in the original SQP+ algorithm is allowed, using the parameter $\epsilon_5 = 10^{-6}$.

$$\phi_\pi(\mathbf{x}_k + \mathbf{d}_k^{iq} + \beta \mathbf{d}_k^{eq}) \leq \phi_\pi(\mathbf{x}_k) - \sigma qred_\pi(\mathbf{d}_k^{iq}) - \epsilon_5.$$

Secondly, if the previous condition is not satisfied (Step 9 in Algorithm 1), the full IQP step ($\alpha = 1$) is taken (up to 5 consecutive iteratione). In practice, the algorithm takes, in most of the iterations, a step direction involving both phases, i.e, $\mathbf{d}_k = \mathbf{d}_k^{iq} + \beta \mathbf{d}_k^{eq}$. However, in very few examples, this descent direction does not reduce the merit function and it stalls in a local minimum with a small KKT error but not sufficiently small for convergence. In those situations the KKT conditions are satisfied when a full inequality step is forced. The SQP+ implemented to compare the performance of TopSQP has the same parameters and structure

Table 3 Parameter settings for Gurobi for solving the IQP sub-problem of TopSQP. The table contains the name of the parameter, a brief description and the value

Parameter	Description	Value
`OptimalityTol`	Optimality tolerance	10^{-9}
`FeasibilityTol`	Feasibility tolerance	10^{-9}
`threads`	Number of OMP threads	4
`presolve`	Presolve level	0 (off)

Table 4 Parameter settings for the interior point implemented for solving the IQP sub-problem of SQP+. The table contains the name of the parameter, a brief description and the value

Parameter	Description	Value
ϵ_1	Stationarity tolerance	10^{-11}
ϵ_2	Feasibility tolerance	10^{-11}
ϵ_3	Complementarity tolerance	10^{-8}
$\mu_{\min}$	Minimum value of the barrier parameter	10^{-9}
τ	Fraction to the boundary parameter	0.995
γ	Constant for the adaptive barrier parameter update scheme (Vanderbei and Shanno 1999)	0.1
r	Constant for the adaptive barrier parameter update scheme (Vanderbei and Shanno 1999)	0.95
`l-memory`	Maximum size of history in BFGS	25

as TopSQP. The only difference is in the IQP phase. A specific purpose interior point method based on Vanderbei and Shanno (1999) is implemented in MATLAB for solving the IQP sub-problem using a limited memory quasi-Newton approximation of the Hessian (Byrd et al. 1996).

Regarding the finite element analysis, the design domain is discretized using the same size of plane stress elements (with 4 nodes per element), and then the element stiffness matrix is the same for all elements. The code of the finite element analysis is based on Andreassen et al. (2011).

8 Numerical Experiments

The specific purpose TopSQP solver is compared to a particular implementation of the general SQP+, the first-order structural topology optimization solver, GCMMA (Svanberg 2002), and two general purpose solvers, SNOPT (Gill et al. 2005) and IPOPT (Wächter and Biegler 2006). These last two solvers (namely SNOPT and IPOPT-N) use limited memory BFGS approximation of the Hessian. Moreover, the best solver (in terms of objective function value) according to Rojas-Labanda and Stolpe (2015), IPOPT-S, is under consideration. IPOPT-S[3] solves the SAND (Simultaneous Analysis and Design, see e.g. Arora and Wang (2005)) formulation using the exact Hessian. More information about these solvers and their specific parameter settings can be found in Rojas-Labanda and Stolpe (2015). The parameter values set for the TopSQP (and SQP+), Gurobi solver (IQP-TopSQP), and the interior point method (IQP-SQP+) are gathered in Tables 2, 3 and 4 respectively. In addition, Table 5 contains the parameter values of the minimum compliance problem used for TopSQP 3 (and SQP+). IPOPT and SNOPT are able to find accurate local minima (small optimality tolerance). Thus, the optimality conditions for IPOPT and SNOPT are set as in TopSQP, i.e. `feas norm` $= 10^{-8}$ and `kkt norm` $= 10^{-6}$. More details of how the KKT norm is obtained in these solvers can be found in Wächter and Biegler (2006) and Gill et al. (2005). The stopping criterion of GCMMA is `kkt norm` $= 10^{-4}$ and `feas norm` $= 10^{-8}$ (first-order method). The maximum number of iterations for all the solvers is set to `max iter` $= 1,000$. See more details in Rojas-Labanda and Stolpe (2015).

All the computations were done on a Intel Xeon e5-2680v2 ten-core processor, running at 2.8GHz with 64 GB RAM. Only Gurobi (IQP phase) runs in parallel using four threads. TopSQP, SQP+, IPOPT, SNOPT, and GCMMA all run in serial.

The numerical experiments to assess the performance of TopSQP are presented using performance profiles, see Dolan and Moré (2002). The specific minimum compliance test set consists of 225 2D medium-size instances (with 400-40,000 finite elements) as defined in Rojas-Labanda and Stolpe (2015). Figure 1 shows how the problem instances are distributed with regards to the number of elements, i.e. size of the problem. Since a time limit is set to 300 hours, the execution of TopSQP (MATLAB general purpose implementation) is not finished[4] for 18 problem instances. SQP+ has the same problem in 10 instances. In addition, IPOPT-S (SAND formulation) has some issues in the linear algebra[5] for 12 problem instances. Nevertheless, the last intermediate design obtained in these instances is considered as the final design in the benchmarking study.

[3] For simplicity, the default linear algebra package MUMPS (Amestoy et al. 2000) is used in both nested (IPOPT-N) and SAND (IPOPT-S) formulation.

[4] The computational time required for the solver is highly dependent on the number of processors, the method, as well as the number of threads used during the execution.

[5] IPOPT needs to reallocate memory and thus, it has difficulties to converge.

Table 5 Values of characteristic parameters of topology optimization problems solved by TopSQP. A small contrast of E_1/E_v is considered since we are mostly interested in the behaviour of the solvers. In addition, the values of E_v and E_1 are chosen to well-scaled problems for the solvers (see Rojas-Labanda and Stolpe (2015) for more details)

Parameter	Description	Value
E_v	Young's modulus value for void material	10^{-1}
E_1	Young's modulus value for solid material	10^2
p	SIMP penalization parameter	3
$r_{\min}$	radius for the density filter (L_x is the length in the x direction)	$0.04L_x$

The performance profiles show the percentage of problems (in the test set) where a solver s obtains different relative ratios of performance (defined with the parameter τ). In other words, for a given solver s, the function ρ_s defined is represented as

$$\rho_s(\tau) = \frac{1}{N}\text{size}\{\tilde{p} \in P : r_{\tilde{p},s} \leq \tau\},$$

or

$$\rho_s(\tau) = \frac{1}{N}\text{size}\{\tilde{p} \in P : \log_{10}(r_{\tilde{p},s}) \leq \tau\}.$$

Here, P is the set of problems with $\tilde{p} \in P$ and N the size of P. The ratio of performance for a solver s for each problem $\tilde{p}$ is defined as

$$r_{\tilde{p},s} = \frac{m_{\tilde{p},s}}{\min\{m_{\tilde{p},s} : s \in S\}},$$

with m a measure of performance, such as

$$m_{\tilde{p},s} = \text{iter}_{\tilde{p},s} = \{\text{ number of iterations required to solve the problem } \tilde{p} \text{ by a solver } s\}.$$

In these numerical experiments, the solvers are compared using four different criteria: the objective function value, the number of iterations, the number of stiffness matrix assemblies, and the computational time. Here, one iteration refers to an outer iteration in GCMMA, an IQP solved in TopSQP and SQP+, an QP solved in SNOPT, and a barrier sub-problem in IPOPT.

Furthermore, at the maximum value ratio r_M the performance profiles reflect the robustness of the solvers. In these numerical experiments, a solver fails if the KKT error is larger than $\omega_{\max} = 10^{-3}$. Thus, the term robustness refers to the capability of obtaining a design with a KKT accuracy lower than or equal to 10^{-3}. More details about the impact

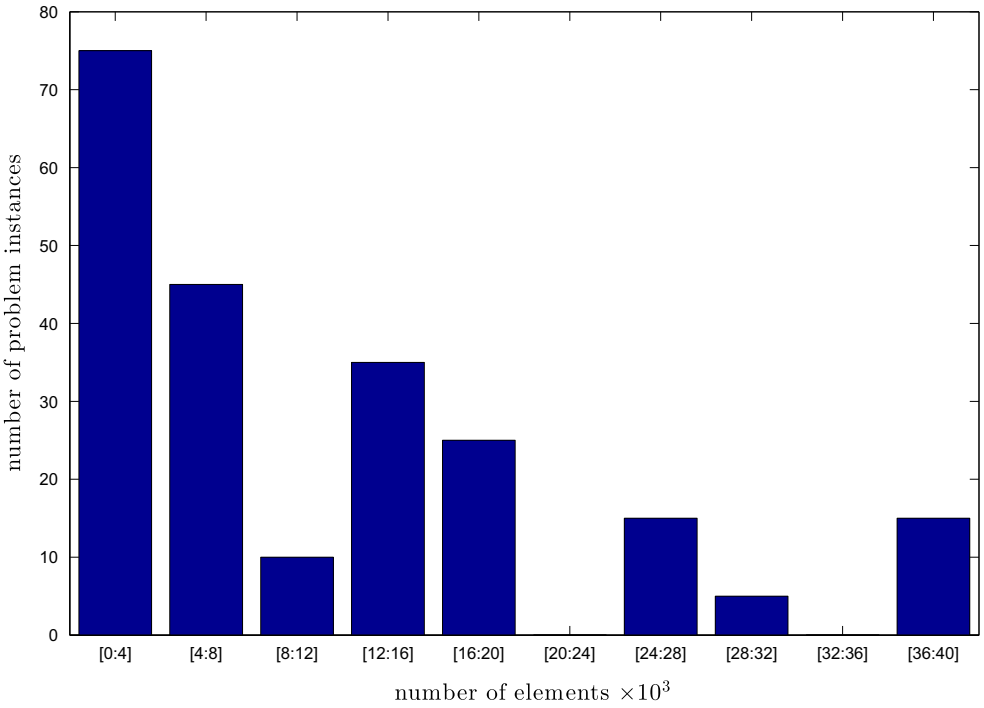
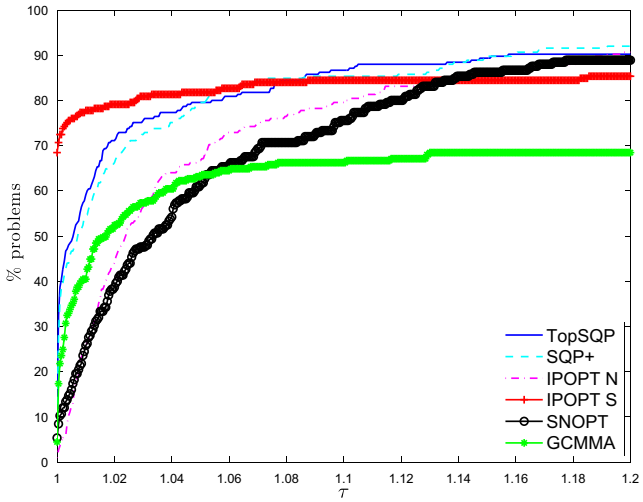
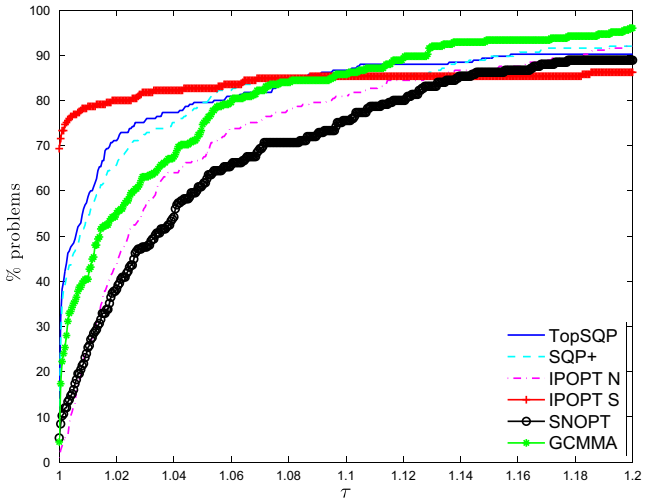


Fig. 1 Distribution of the number of instances of the test set for different number of elements ranges



(a) Objective function value ($\omega_{\max} = 10^{-3}$).



(b) Objective function value ($\omega_{\max} \gg 0$).

Fig. 2 Performance profile for a test set of 225 minimum compliance problems. The performance is measured by the objective function value. Figure 2a shows the performance when designs with KKT error higher than $\omega_{\max} = 10^{-3}$ are penalized. Figure 2b shows the performance without any penalization measured, i.e., with $\omega_{\max} \gg 0$

of this threshold can be found in Rojas-Labanda and Stolpe (2015).

Figure 2a shows the performance profile for the objective function value. IPOPT-S still has the highest number of problems where the designs have the minimum objective function value (70 % of the cases). Nevertheless, when τ is relatively small, ($\tau = 1.07$), the performance of both TopSQP and SQP+ are the same as IPOPT-S with a success of 81 %. TopSQP (and SQP+) outperforms the other solvers in terms of objective function value for τ close to 1. Although SQP+ uses a BFGS approximation in the IQP, thanks to the use of the EQP phase, the objective function value is almost identical to TopSQP.

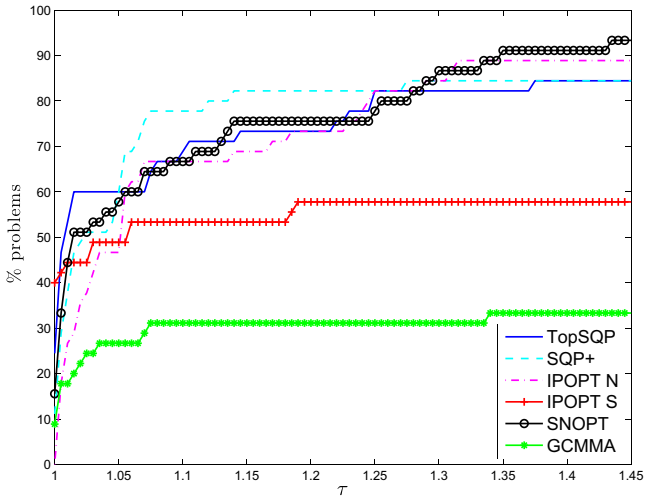
Some of the methods presented in the study produce feasible iterates. Although the first-order optimality conditions are not satisfied (KKT error higher than $\omega_{\max}$), the objective function values of feasible designs might still be acceptable. Figure 2 shows the performance profile (for objective function value) when no penalization in the performance profiles is applied. The difference is meaningful for GCMMA. In general, its feasible iterates produce reasonably good designs. Although, the performance of IPOPT-S and TopSQP is still better, in general all the solvers produce designs with similar objective function values. In 80 % of the problems, the difference of objective function values is smaller than 12 % (i.e. $\tau = 1.12$).

In almost the 30 % of the problems solved with GCMMA, the KKT error is higher than 10^{-3}. Before analysing the performance of the methods for different criteria, it is important to study in detail why GCMMA has difficulties to converge. Figure 3 shows the performance profiles (for objective function value) for smaller test sets of problems. The original test set is partitioned into five. Each subset gathers 45 problems in which the volume fraction is the same ($V = 0.1$, 0.2, 0.3, 0.4, and 0.5). It seems clear that GCMMA has difficulties solving problems with low volume fractions, see Fig. 3a, b in which the test set collects problems with $V = 0.1$ and $V = 0.2$, respectively.

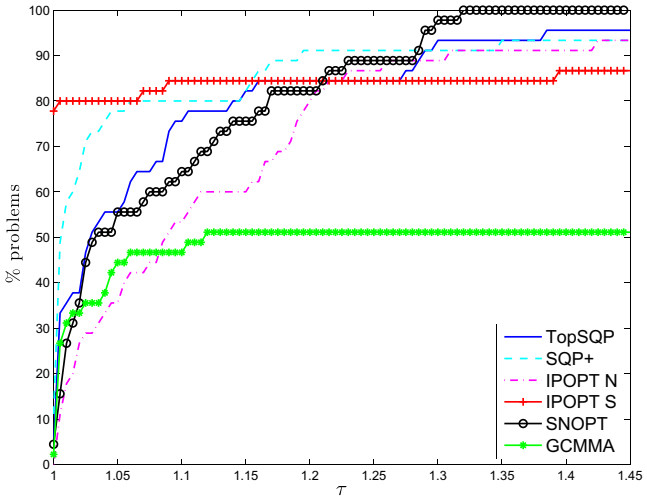
Regarding the performance for the number of iterations (Fig. 4a), TopSQP produces designs using the smallest number of iterations, with a very similar performance to SNOPT. However, the designs for the latter solver have worse objective function values (see Fig. 2a). Due to the use of BFGS approximations, the number of SQP+ iterations is considerably larger than in TopSQP.

Since TopSQP has two phases, it is also important to compare the performance using other criteria to check the cost of every major iteration. Figure 4b shows the performance profiles when the solvers are compared with the number of stiffness matrix assemblies, which is equivalent to the number of function evaluations. The few number of stiffness matrix assembled for TopSQP is outstanding. In contrast to SNOPT or IPOPT-S, TopSQP usually evaluates the stiffness matrix once per iteration. This is due to the definition of the line search.

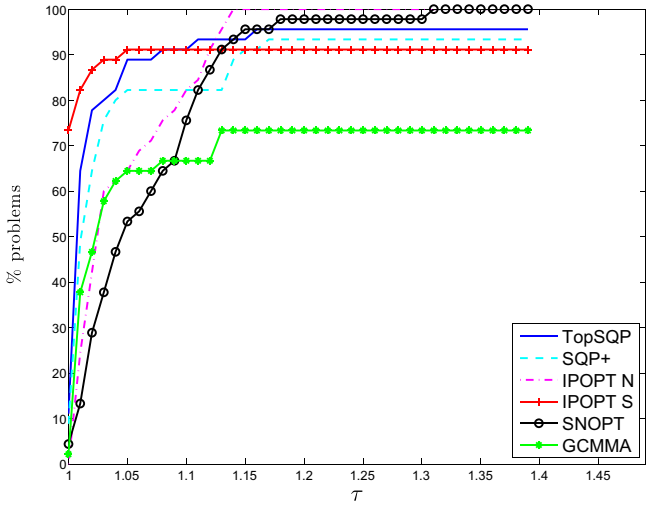
A very important aspect in the comparison of solvers is the computational time required for obtaining a solution. On the other hand, we need to be cautious since the solvers have different interfaces, they are programmed in different languages, and can be linked to different linear algebra packages. The computational time could be highly affected by this. Although it is preferable to compare the computational



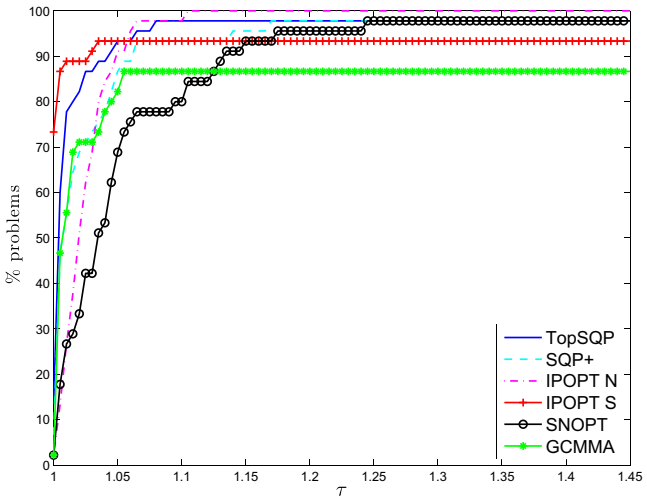
(a) Objective function value ($V = 0.1$)



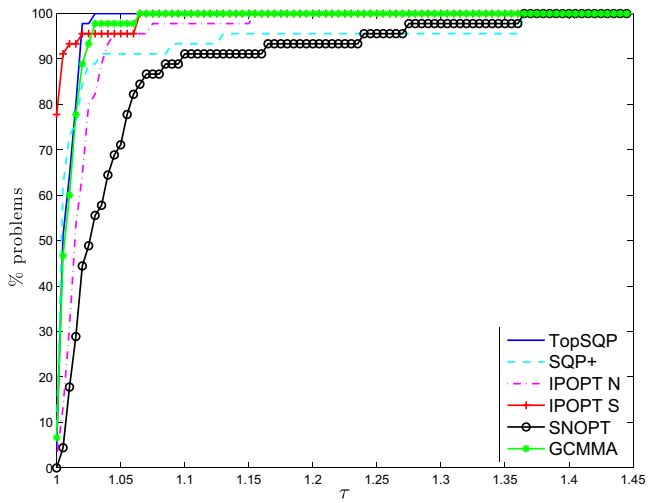
(b) Objective function value ($V = 0.2$)



(c) Objective function value ($V = 0.3$)

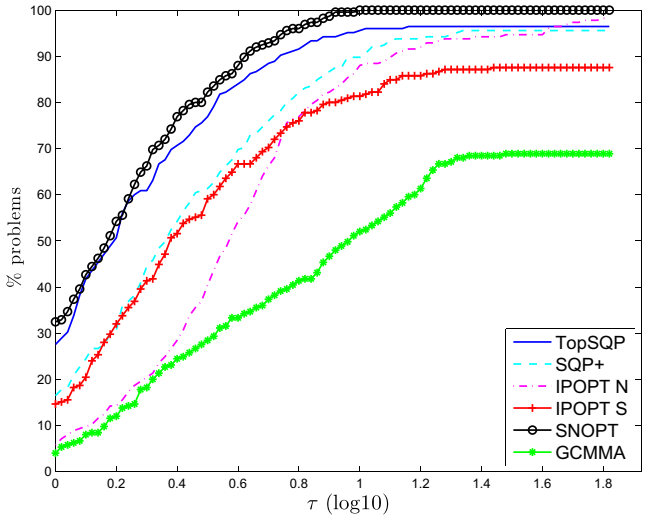


(d) Objective function value ($V = 0.4$)

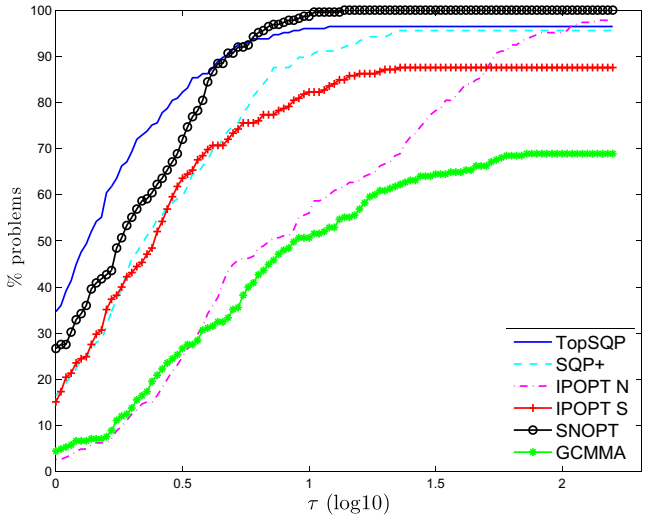


(e) Objective function value ($V = 0.5$)

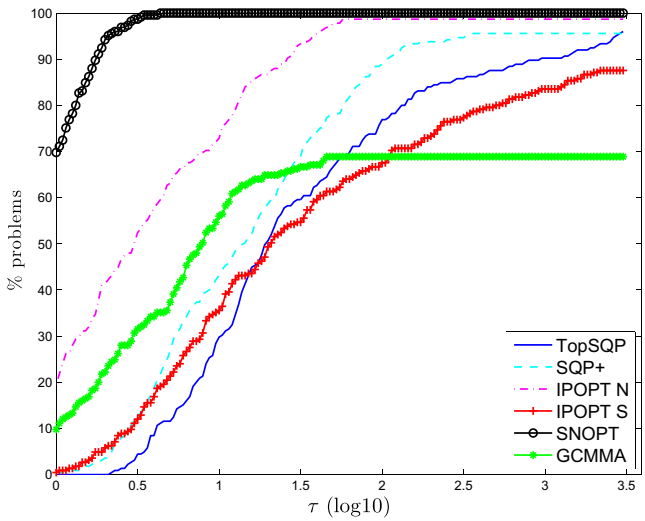
Fig. 3 Performance profiles for five different subsets of the minimum compliance problems (45 problem instances each). The problems are divided depending on their volume fraction. The performance is measured by the objective function value



(a) Number of iterations



(b) Stiffness matrix assemblies



(c) Computational time

Fig. 4 Performance profiles for a test set of 225 minimum compliance problems. The performance is measured by the objective function value (2a), the number of iterations (4a), the number of stiffness matrix assemblies (4b) and the computational time (4c)

cost of the solvers using a more objective criterion such as the number of iterations or the number of stiffness matrix assemblies, it is remarkable the amount of time required for TopSQP (Fig. 4c). It performs slightly worse than IPOPT-S (SAND formulation). The major amount of time in the proposed method is spent in the IQP sub-problem. More sophisticated and advanced convex quadratic methods must be developed to solve this sub-problem in order to produce an efficient and fast method. Nevertheless, both SQP+ and TopSQP are considered a good compromise between accurate and good designs (good objective function values) and computational time. In contrast, SQP+ requires more iterations than TopSQP, thus the use of information based on the Hessian is essential to reduce the number of iterations.

Finally, TopSQP, SQP+, IPOPT-N, and SNOPT have excellent robustness properties, i.e. the KKT error is lower than 10^{-3} in about the 96 % of the test set. In contrast, the robustness of GCMMA is highly dependent on the problems. GCMMA performs very well in terms of the objective function value when the volume fraction is large, with a robustness of 100 % (see Fig. 3e). On the other hand, the robustness of GCMMA drops drastically to 30 % for volume fraction equal to $V = 0.1$.

The use of efficient second-order methods (such as IPOPT-S and TopSQP) is essential in topology optimization to produce accurate designs[6] in few iterations. The

[6]Designs with good objective function values and low KKT error.

results confirm that they are better not only than the classical methods but also than other (second-order) methods where the Hessian is approximated using BFGS (SNOPT and IPOPT-N). Only IPOPT-S beats TopSQP in objective function value, although at a small ratio of performance TopSQP and SQP+ are as competent as IPOPT-S in this aspect. In contrast, TopSQP requires fewer function evaluations than SQP+ and IPOPT-S. In addition, TopSQP and SQP+ have the benefits of solving the nested formulation, for instance it has feasible solutions at intermediate steps, less number of variables, and less memory usage.

9 Limitations of TopSQP in structural topology optimization

The main advantage of the IQP phase for minimum compliance problems is the ease of finding a good positive semi-definite approximation of the Hessian, for which the dual is much faster to solve than the primal optimization problem. Moreover, the improvement of the EQP relies on the symmetry of the approximate Hessian. However, these benefits are not satisfied for all classes of structural topology optimization problems. For instance, in compliant mechanism design problems (Bendsøe and Sigmund 2003; Sigmund 1997), the exact Hessian is

$$\begin{aligned}\nabla^2\mathcal{L}(\mathbf{t},\boldsymbol{\lambda}) \;=\; & \mathbf{F}(\mathbf{t},\boldsymbol{\lambda})^T\mathbf{K}^{-1}(\mathbf{t})\mathbf{F}(\mathbf{t},\mathbf{u}) \\ & +\mathbf{F}(\mathbf{t},\mathbf{u})^T\mathbf{K}^{-1}(\mathbf{t})\mathbf{F}(\mathbf{t},\boldsymbol{\lambda}) - \mathbf{Q}(\mathbf{t},\boldsymbol{\lambda},\mathbf{u})\end{aligned}$$

where

$$\mathbf{Q}(\mathbf{t},\boldsymbol{\lambda},\mathbf{u}) = \operatorname{diag}(\boldsymbol{\lambda}^T(\mathbf{t})\frac{\partial^2\mathbf{K}_i(t_i)}{\partial t_i^2}\mathbf{u}(\mathbf{t}))$$

$$\mathbf{F}(\mathbf{t},\mathbf{u}) = \left(\frac{\partial\mathbf{K}_1(t_1)}{\partial t_1}\mathbf{u}(\mathbf{t}) \cdots \frac{\partial\mathbf{K}_n(t_n)}{\partial t_n}\mathbf{u}(\mathbf{t})\right).$$

and λ the adjoint variable used for the sensitivity analysis.

For this class of problems, the Hessian is more difficult to approximate. It is possible to use simple approximations, but the convergence rate could drop significantly.

Nevertheless, the major limitation of the proposed TopSQP, even for minimum compliance problems, is the computational time required to obtain the optimal solution of the sub-problems. Most of this time is spent in the IQP phase, where a quadratic convex sub-problem is solved. Efficient linear solvers, such as iterative methods are essential to use TopSQP for large-scale topology optimization problems.

10 Conclusions and further research

An efficient second-order sequential quadratic programming method based on SQP+ from Morales et al. (2010) is presented for topology optimization problems. More specifically, the minimum compliance problem is solved using information of the Hessian in both, the IQP and EQP phases. An efficient approximate Hessian is used in both phases, producing good estimates of the search direction. These sub-problems are efficiently reformulated taking advantages of the structure of the problem.

The numerical experiments confirm the benefits of using second-order information not only for reducing the number of iterations but also for decreasing the objective function values. IPOPT-S, SQP+, and TopSQP are the most competent methods to produce designs with good objective function value. The comparison of the performance between IPOPT-S and TopSQP (exact Hessian) and SNOPT and IPOPT-N (BFGS approximation), reinforces that the use of information based on the exact Hessian is important to obtain good designs.

Since SQP+ uses information of the Hessian in the EQP phase, the designs are as good as TopSQP, although SQP+ requires more iterations due to the poorer estimation in the IQP phase.

Additionally, TopSQP outperforms GCMMA not only in the objective function value but also in the number of iterations, the number of stiffness matrix assemblies, and the overall robustness.

Although IPOPT-S outperforms the other solvers when measuring the objective function value, all the solvers are able to produce similar results. Indeed the objective function value of TopSQP is very close to the minimum possible. In contrast, IPOPT-S requires more iterations and function evaluations than TopSQP. Finally, TopSQP solves the nested formulation with all the advantages that brings with it.

Further work must be done in order to extend and generalize the proposed TopSQP method. Future work is directed to solve more general topology optimization problems such as compliant mechanism design problems and problems with displacement constraints. Additional investigations are also required to extend the code to be able to solve large-scale problems.

Acknowledgments We would like to thank Professor Krister Svanberg at KTH in Stockholm for providing the implementation of GCMMA. We extend our sincere thanks to two reviewers and the editor for providing many useful comments and suggestions that improved the quality of this article.

References

Amestoy PR, Duff IS, L'Excellent JY (2000) Multifrontal parallel distributed symmetric and unsymmetric solvers. Comput Methods Appl Mech Eng 184(2–4):501–520

Andreassen E, Clausen A, Schevenels M, Lazarov BS, Sigmund O (2011) Efficient topology optimization in MATLAB using 88 lines of code. Struct Multidiscip Optim 43(1):1–16

Arora JS, Wang Q (2005) Review of formulations for structural and mechanical system optimization. Struct Multidiscip Optim 30(4):251–272
Bendsøe MP (1989) Optimal shape design as a material distribution problem. Struct Optim 1(4):192–202
Bendsøe MP, Sigmund O (2003) Topology optimization: Theory, methods and applications. Springer
Boggs PT, Tolle JW (1984) A family of descent functions for constrained optimization. SIAM J Numer Anal 21(6):1146–1161
Boggs PT, Tolle JW (1995) Sequential Quadratic Programming. Acta Numer 4:1–51
Bourdin B (2001) Filters in topology optimization. Int J Numer Methods Eng 50(9):2143–2158
Boyd S, Vandenberghe L (2010) Convex Optimization. Cambridge University Press
Byrd RH, Nocedal J, Schnabel RB (1996) Representations of quasi-Newton matrices and their use in limited memory methods. NAM-03, Technical Report, Northwestern University, Department of Electrical Engineering and Computer Science
Byrd RH, Gould NIM, Nocedal J, Waltz RA (2004) An algorithm for nonlinear optimization using linear programming and equality constrained subproblems. Math Program 48:27–48
Conn AR, Gould NIM, Toint PL (1987) Trust Region Methods. Society for Industrial and Applied Mathematics
Curtis FE, Nocedal J (2008) Flexible penalty functions for nonlinear constrained optimization. IMA J Numer Anal 25(4):749–769
Curtis FE, Johnson TC, Robinson DP, Wächter A (2014) An inexact sequential quadratic optimization algorithm for nonlinear optimization. SIAM J Optim 24(3):1041–1074
Dennis JE, Moré JJ (1977) Quasi-newton methods, motivation and theory. SIAM Rev 19(1):46–89
Dolan ED, Moré JJ (2002) Benchmarking optimization software with performance profiles. Math Program 91(2):201–213
Dreyer T, Maar B, Schulz V (2000) Multigrid optimization in applications. J Comput Appl Math 120(1-2):67–84
Etman LF, Groenwold AA, Rooda JE (2012) First-order sequential convex programming using approximate diagonal QP subproblems. Struct Multidiscip Optim 45(4):479–488
Fletcher R, Leyffer S (1998) User manual for filterSQP. Technical Report NA/181, University of Dundee Numerical Analysis
Fletcher R, Leyffer S (2002) Nonlinear programming without a penalty function. Math Program 91(2):239–269
Fleury C (1989a) CONLIN: An efficient dual optimizer based on convex approximation concepts. Struct Optim 1(2):81–89
Fleury C (1989b) Efficient approximation concepts using second order information. Int J Numer Methods Eng 28(9):2041–2058
Fleury C (1989c) First and second order convex approximation strategies in structural optimization. Struct Optim 1(1):3–10
Forsgren A (2002) Inertia-controlling factorizations for optimization algorithms. Appl Numer Math 43(1-2):91–107
Forsgren A, Gill PE (1998) Primal-dual interior methods for nonconvex nonlinear programming. SIAM J Optim 8(4):1132–1152
Forsgren A, Murray W (1997) Newton methods for large-scale linear inequality-constrained minimization. SIAM J Optim 7(1):162–176
Gill PE, Robinson DP (2013) A globally convergent stabilized SQP method. SIAM J Optim 23(4):1983–2010
Gill PE, Wong EL (2012) Sequential quadratic programming methods. Mixed Integer Nonlinear Program, IMA Vol Math Appl 154:147–224
Gill PE, Wong EL (2014) Convexification schemes for SQP methods. Technical Report CCoM 14-6 Center for Computational Mathematics. University of California, San Diego
Gill PE, Murray W, Saunders MA, Wright MH (1998) User's guide for NPSOL 5.0: A fortran package for nonlinear programming. Technical Report, Systems Optimization Laboratory, Department of Opertations Research, Stanford University
Gill PE, Murray W, Saunders MA (2005) SNOPT: An SQP algorithm for large-scale constrained optimization. SIAM J Optim 47(4):99–131
Gould NIM (1985) On practical conditions for the existence and uniqueness of solutions to the general equality quadratic programming problem. Math Program 32(1):90–99
Gould NIM, Robinson DP (2010) A second derivative SQP method: Global convergence. SIAM J Optim 20(4):2023–2048
Gurobi Optimization (2013) Gurobi Optimizer Reference Manual - Version 5.6. Technical Report, Gurobi Optimization Inc., www.gurobi.com/documentation/5.6/refman.pdf
Hager WW (1999) Stabilized sequential quadratic programming. In: Computational Optimization. Springer, pp 253–273
Higham NJ, Cheng SH (1998) Modifying the inertia of matrices arising in optimization. Linear Algebra Appl 275–276:261–279
IBM Corporation (2014) IBM ILOG CPLEX Optimization Studio V12.6.0 documentation. Technical Report., IBM Corporation, www-01.ibm.com/support/knowledgecenter/SSSA5P_12.6.0
Leyffer S, Mahajan A (2010) Software for Nonlinearly Constrained Optimization. Technical Report ANS/MCS-P1768-0610, Mathematics and Computer Science Division, Argonne National Laboratory
Luenberger DG, Ye Y (2008) Linear and Nonlinear Programming. Springer
Manyem P, Ugon J (2012) Computational Complexity, NP Completeness and Optimization Duality: A survey. Technical Report, Electronic Colloquium on Computational Complexity, Report 9
Morales JL, Nocedal J, Wu Y (2010) A sequential quadratic programming algorithm with an additional equality constrained phase. J Numer Anal 32(2):553–579
Moré JJ, Thuente DJ (1994) Line search algorithms with guaranteed sufficient decrease. ACM Trans Math Softw 20(3):286–307
Murty KG, Kabadi SN (1987) Some NP-complete problems in quadratic and nonlinear programming. Math Program 39:117–129
Nocedal J, Wright SJ (1999) Numerical Optimization. Springer
Orozco CE, Ghattas ON (1997) A reduced SAND method for optimal design of nonlinear structures. Int J Numer Methods Eng 40(15):2759–2774
Prieto FJ (1989) Sequential quadratic programming algorithms for optimization. Technical Report SOL 89-7, Systems Optimization Laboratory, Department of Operations Research. Stanford University
Rojas-Labanda S, Stolpe M (2015) Benchmarking optimization solvers for structural topology optimization. Struct Multidiscip Optim 52(3):527–547
Rozvany GIN, Zhou M, Birker T (1992) Generalized shape optimization without homogenization. Struct Optim 4(3–4):250–252
Schittkowski K (2002) NLPQLP : A new fortran implementation of a Sequential Quadratic Programming algorithm. Technical Report, Department of Mathematics. University of Bayreuth
Shen C, Xue W, Chen X (2010) Global convergence of a robust filter SQP algorithm. Eur J Oper Res 206(1):34–45
Sigmund O (1997) On the design of compliant mechanisms using topology optimization. J Struct Mech 25(4):492–526
Sigmund O, Petersson J (1998) Numerical instabilities in topology optimization: A survey on procedures dealing with checkerboards, mesh-dependencies and local minima. Struct Optim 16(2):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

The MathWorks (2014) Optimization Toolbox User's Guide R 2014 a

Vanderbei RJ, Shanno DF (1999) An interior-point algorithm for nonconvex nonlinear programming. Comput Optim Appl 13 (1-3):231–252

Wächter A, Biegler LT (2006) On the implementation of an interior point filter line-search algorithm for large-scale nonlinear programming. Math Program 106(1):25–57

Wang B, Pu D (2013) Flexible penalty functions for SQP algorithm with additional equality constrained phase. Technical Report. In: Proceedings of the 2013 International Conference on Advance Mechanic System, China, pp 25–27

Zhou M, Rozvany GIN (1991) The COC algorithm, Part II: Topological, geometrical and generalized shape optimization. Comput Methods Appl Mech Eng 89(1–3):309–336