A first-order equivalent static loads algorithm for optimization of nonlinear static response

Stolpe, Mathias; Pollini, Nicolò

Published in:
Advances in Engineering Software

Link to article, DOI:
10.1016/j.advengsoft.2023.103462

Publication date:
2023

Document Version
Publisher's PDF, also known as Version of record

Citation (APA):
Stolpe, M., & Pollini, N. (2023). A first-order equivalent static loads algorithm for optimization of nonlinear static response. *Advances in Engineering Software*, *182*, Article 103462.
https://doi.org/10.1016/j.advengsoft.2023.103462

Contents lists available at ScienceDirect

Advances in Engineering Software

journal homepage: www.elsevier.com/locate/advengsoft

A first-order equivalent static loads algorithm for optimization of nonlinear static response

Mathias Stolpe [*], Nicolò Pollini [1]

Technical University of Denmark (DTU), DTU Wind and Energy Systems, Frederiksborgvej 399, 4000 Roskilde, Denmark

ARTICLE INFO

Keywords:
Structural optimization
Sensitivity analysis
Equivalent static load
Nonlinear response

ABSTRACT

The Equivalent Static Loads (ESL) algorithm for nonlinear static response structural optimization is modified to promote convergence to designs satisfying first-order optimality conditions. The modifications involve first-order estimates of the equivalent static loads considered in the sub-problems and algorithmic stabilization through a trust-region approach. The practical convergence properties of the original and modified algorithms are assessed through numerical experiments on a set of reproducible structural size optimization problems. The results demonstrate the capabilities of the proposed algorithm in finding optimized designs that satisfy first-order optimality conditions numerically with modest computational resources.

1. Introduction

Equivalent Static Loads (ESL) algorithms are a family of structural optimization approaches that are becoming increasingly popular. Their main motivation is the possibility of reducing computational complexity and cost of given structural optimization instances. ESL algorithms approximate given nonlinear or transient structural optimization problems with sequences of linear static optimization problems. In the context of nonlinear static response structural optimization, an ESL algorithm was first introduced in [1]. The main idea is to calculate a vector of equivalent loads $\mathbf{f}_{\mathrm{eq}}$ that produce displacements of the linear structure $\mathbf{K}_l$ equal to those obtained with nonlinear structural analysis $\mathbf{u}_{nl}$ through

$$\mathbf{f}_{\mathrm{eq}} = \mathbf{K}_l \, \mathbf{u}_{nl} \tag{1}$$

These loads are then used for optimizing the design while considering linear structural behaviour. Once the new design is obtained, new equivalent loads are computed, and the process is repeated until certain convergence criteria are satisfied. The ESL approach described in [1] is further reviewed in [2], and in [3] it is applied to a set of case studies. A software framework for structural optimizations for linear dynamic, non-linear static, and non-linear dynamic responses is discussed in [4]. The ESL algorithm for nonlinear static response presented in [1] builds on similar techniques as proposed for linear dynamic response optimization in e.g. [5,6], and [7]. Moreover, in [8] the ESL algorithm is applied to shape optimization of linear structures subjected to dynamic loads. ESL is also applied to nonlinear dynamic response optimization for the design of car roof [9] and frontal [10] structural components. The application of ESL to the optimization of multi-body dynamic systems is discussed in [11]. More recently, topology optimization of structures with nonlinear dynamic behaviour has been discussed in [12]. The optimization of a front hood for nonlinear dynamic structural response based on the ESL algorithm is discussed in [13,14]. The ESL algorithm has been applied also to the car roof crash test optimization problem in [15].

This paper has the following major objectives and contributions:

- To discuss the numerical behaviour of ESL as proposed in the literature, with particular emphasis on its capabilities of obtaining designs that satisfy necessary first-order optimality conditions, i.e. the KKT conditions [16];
- To propose, and numerically study, a class of problems for which the linear response sub-problems are equivalent to convex problems and thus can be solved to proven global optimality;
- To suggest a modification of the original ESL algorithm for static problems based on nonlinear structural analysis by changing the sub-problems in a way that satisfies first-order approximation properties at every (outer) iteration;
- Lastly, to assess the proposed ESL algorithm, namely F-ESL, by studying its numerical convergence behaviour.

The ESL algorithm in its original formulation does not have a guaranteed capability of reaching, or recognizing, final designs that satisfy necessary first-order optimality conditions. This limitation has already

[*] Corresponding author.
E-mail addresses: matst@dtu.dk (M. Stolpe), nicolo@technion.ac.il (N. Pollini).
[1] NP is now a Horev Fellow at the Technion - Israel Institute of Technology, Faculty of Civil and Environmental Engineering, Haifa, Israel.

https://doi.org/10.1016/j.advengsoft.2023.103462
Received 19 August 2022; Received in revised form 16 March 2023; Accepted 23 March 2023
Available online 6 May 2023

been discussed in the literature for the case of linear transient structural optimization problems [17,18]. An alternative to the original formulation of ESL for nonlinear transient structural optimization problems, a difference-based extension of the ESL algorithm (DiESL), has been recently proposed in [19]. The main idea in DiESL is to compute incremental equivalent static loads by considering incremental displacements and the structural stiffness matrix of the displaced structure in each time-step. DiESL is further extended in [20] by considering heterogeneous time steps, adaptively placed at points in time where nonlinearities are dominant.

In this paper, instead, we propose a modification of the original ESL to ensure that the sub-problems satisfy first-order approximation properties for nonlinear static response problems. The result is the F-ESL algorithm. For an optimization algorithm having sub-problems satisfying first-order approximation properties is fundamental. This property is present in common optimization algorithms such as SQP [16] and sequential convex approximation algorithms such as MMA [21]. If the linear static response sub-problems of a ESL algorithm are not first-order approximations of the original problem, it is possible that the optimization solver used in the sub-problems does not recognize the optimality for a point that does satisfy the first-order optimality conditions. As a result, the optimization algorithm could proceed towards search directions that lead away from actual optimal solutions.

Nonlinear response optimization for truss structures has been extensively discussed in the literature, e.g. [22–27]. The numerical results in this work are entirely based on truss and frame optimization problems for a number of reasons. The main reason being to promote the reproducibility of the results. The implementations are based on standard truss and frame analysis, analytical design sensitivity analysis, and problem formulations that are extensively studied in the literature. The second reason is that for the truss examples[2] it becomes possible to utilize well-known convex reformulations of what otherwise would have been non-convex problems.

One of the properties of the ESL algorithm is that the sub-problems, with few exceptions, are non-convex. It is therefore not possible, in practice, to ensure that with ESL the sub-problems are solved to global optimality. As a result, during the optimization process it may be challenging to ensure the decrease of a merit function to promote convergence to a point satisfying first-order optimality conditions. Non-convexity is, of course, a common property of many structural optimization problems, even for problems based on linear structural behaviour. However, in one of the numerical applications discussed herein the sub-problems of the equivalent static loads algorithm are either convex or can be reformulated as convex problems. The sub-problems can thus be solved to proven global optimality by standard numerical optimization methods that provide accurate numerical solutions. The convexity properties, for these problems, require linear structural analysis combined with stiffness matrices that are linear in the design variables. The first condition is generally not satisfied for the original problem instances as they are based on nonlinear analysis.

The numerical experiments are designed to avoid several of the complications associated with industrial and large-scale problems resulting from real-life applications. The decisions underlying this paper are intended to ensure, to the largest extent possible, that the fundamental convergence properties of the equivalent static loads algorithm are in focus. We therefore purposely avoid large-scale problems to avoid long computation times. We also avoid complicated geometries and advanced finite elements which could result in difficulties with e.g. accuracy in the structural analysis and sensitivity analysis. Another reason is to avoid issues with reproducibility of the results due to confidentiality and lack of model data availability. Finally, we do not consider optimization problems with excessively complicated problem formulations that could lead to sub-problems with disadvantageous properties such as infeasibility, or lack of regularity.

In what follows, Section 2 starts with a re-statement of the basic equivalent static loads algorithm for static nonlinear response optimization as proposed in [1] (Section 2.1). Thereafter follows the statement of the first modifications of the sub-problem in the algorithm and the theoretical consequences thereof (Section 2.2). This refers particularly to the approximation properties of the sub-problems. The algorithm is then further modified in Section 3. We attempt to stabilize the algorithm by a trust-region approach intended to promote convergence from arbitrary starting points. An application where the sub-problems are equivalent to convex problems is presented in some detail in Section 4. The details of the nonlinear structural formulations considered are given in Section 5. In Section 6 the theoretical arguments are supported by a number of numerical experiments with the aim of studying the practical convergence properties of the basic (ESL) and modified (F-ESL) algorithms. The numerical results are discussed in Section 7, and final conclusions are drawn in Section 8.

2. General problem formulations and algorithms

A general structural optimization problem with continuous design variables $\mathbf{v} \in \mathbb{R}^n$ can be stated as

$$\begin{array}{lll} \underset{\mathbf{v}\in\mathbb{R}^n}{\text{minimize}} & f(\mathbf{v},\mathbf{u}(\mathbf{v})) & \\ \text{subject to} & c_i(\mathbf{v},\mathbf{u}(\mathbf{v})) \leq 0 & i=1,\dots,m \\ & \mathbf{v}^{\min} \leq \mathbf{v} \leq \mathbf{v}^{\max} & \end{array} \tag{P}$$

where the state vector $\mathbf{u}(\mathbf{v}) = (\mathbf{u}_1(\mathbf{v})^T \cdots \mathbf{u}_L(\mathbf{v})^T)^T \in \mathbb{R}^{dL}$ solves the governing equations

$$\mathbf{r}_\ell(\mathbf{v},\mathbf{u}_\ell) = \mathbf{0} \;\forall\; \ell = 1,\dots,L \tag{2}$$

and d and L are the number of degrees of freedom and load cases, respectively. The vector with the residuals of the equilibrium equations for the structural analysis for load case ℓ is

$$\mathbf{r}_\ell(\mathbf{v},\mathbf{u}_\ell) = \mathbf{f}_\ell^{\text{int}}(\mathbf{v},\mathbf{u}_\ell) - \mathbf{f}_\ell^{\text{ext}}$$

where $\mathbf{f}_\ell^{\text{int}}$ is the internal load. The external load $\mathbf{f}_\ell^{\text{ext}}$ is assumed to be both design- and state-independent. The model (P) is general enough to model objective $f(\mathbf{v},\mathbf{u}(\mathbf{v}))$ and constraint functions $c_i(\mathbf{v},\mathbf{u}(\mathbf{v}))$ only involving the design variables and not the state vectors and vice versa.

2.1. Basic ESL algorithm for static nonlinear response optimization

The equivalent static loads algorithm is based on solving a sequence of optimization sub-problems. These are in themselves structural optimization problems, but based on different structural analysis assumptions. The sub-problems are parameterized by the current outer iterate $\mathbf{v}^k$ through the equilibrium equations. The sub-problem proposed in e.g. [1,4] for static nonlinear response optimization at outer iteration k is

$$\begin{array}{lll} \underset{\mathbf{v}\in\mathbb{R}^n}{\text{minimize}} & f(\mathbf{v},\mathbf{t}(\mathbf{v};\mathbf{v}^k)) & \\ \text{subject to} & c_i(\mathbf{v},\mathbf{t}(\mathbf{v};\mathbf{v}^k)) \leq 0 & i=1,\dots,m \\ & \mathbf{v}^{\min} \leq \mathbf{v} \leq \mathbf{v}^{\max} & \end{array} \tag{P^k}$$

where the state vector $\mathbf{t}(\mathbf{v};\mathbf{v}^k) = (\mathbf{t}_1(\mathbf{v};\mathbf{v}^k)^T \cdots \mathbf{t}_L(\mathbf{v};\mathbf{v}^k)^T)^T \in \mathbb{R}^{dL}$ solves the linear analysis equations

$$\mathbf{K}(\mathbf{v})\mathbf{t}_\ell(\mathbf{v};\mathbf{v}^k) = \mathbf{f}_{\text{eq},\ell}(\mathbf{v}^k) \;\forall\; \ell = 1,\dots,L \tag{3}$$

where $\mathbf{K}(\mathbf{v})$ is the linear structural analysis stiffness matrix. In the basic formulation of the equivalent static load algorithm, the equivalent load vectors $\mathbf{f}_{\text{eq},\ell}(\mathbf{v})$ are chosen such that at the current outer iterate $\mathbf{v}^k$, they produce the same displacement vector for both linear and nonlinear analysis. This is achieved by first performing nonlinear analysis and then evaluating

$$\mathbf{f}_{\text{eq},\ell}(\mathbf{v}^k) = \mathbf{K}(\mathbf{v}^k)\mathbf{u}_\ell(\mathbf{v}^k) \;\forall\; \ell = 1,\dots,L \tag{4}$$

[2] For one of the problems the truss assumption is not required, but it is convenient.

Remark 1. The notation $\mathbf{u}(\mathbf{v})$ refers to state vectors, which for these problems are functions of the design variables $\mathbf{v}$, that are solutions to the governing equations for the nonlinear analysis, i.e. (2). The notation $\mathbf{t}(\mathbf{v};\mathbf{v}^k)$ refers to solutions to the governing equations for the linear analysis with design variables $\mathbf{v}$ and equivalent loads evaluated at the design defined by $\mathbf{v}^k$, i.e. (3). By definition, $\mathbf{u}(\mathbf{v}^k) = \mathbf{t}(\mathbf{v}^k;\mathbf{v}^k)$ but in general $\mathbf{u}(\mathbf{v}) \neq \mathbf{t}(\mathbf{v};\mathbf{v}^k)$ when $\mathbf{v} \neq \mathbf{v}^k$.

Remark 2. The number of degrees of freedom in the states remain the same between the two analysis situations and the two optimal design problems (P) and (P^k), respectively. The possibility to change the number of degrees of freedom in the sub-problem is not considered.

The basic equivalent static load algorithm for nonlinear response structural optimization, similar to the algorithm from [1], is presented in Algorithm 1 below. The differences between the algorithm presented here and the algorithm in [1] relate to the stopping conditions which have been changed from $\|\mathbf{f}_\ell(\mathbf{v}^k) - \mathbf{f}_\ell(\mathbf{v}^{k-1})\| < \epsilon_o$ to $\|\mathbf{v}^k - \mathbf{v}^{k-1}\| < \epsilon_o$. This change is also suggested in e.g. [28] for ESL algorithms applied to linear dynamic response optimization problems.

Algorithm 1: Basic (ESL) and first-order (F-ESL) equivalent static loads algorithms for nonlinear response structural optimization problems.

```
Initialize $\mathbf{v}^0 \in F$. Initialize the tolerance $\epsilon_o > 0$.
Set the outer iteration counter $k = 0$ and the flag continue = true.
while continue do
    Perform nonlinear structural analysis with $\mathbf{v} = \mathbf{v}^k$ resulting in $\mathbf{u}(\mathbf{v}^k)$.
    Compute the equivalent static loads:
        ESL: $\mathbf{f}_{\mathrm{eq},\ell}(\mathbf{v}^k) = \mathbf{K}(\mathbf{v}^k)\,\mathbf{u}_\ell(\mathbf{v}^k)$
        F-ESL: $\mathbf{f}_{\mathrm{eq},\ell}(\mathbf{v};\mathbf{v}^k) = \mathbf{K}(\mathbf{v}^k)\,\mathbf{u}_\ell(\mathbf{v}^k) + \nabla_{\mathbf{v}}\left(\mathbf{K}(\mathbf{v})\,\mathbf{u}_\ell(\mathbf{v})\right)^T\Big|_{\mathbf{v}^k}(\mathbf{v}-\mathbf{v}^k)$
    if $k > 1$ and $\|\mathbf{v}^k - \mathbf{v}^{k-1}\| < \epsilon_o$ then
        Set continue = false.
    else
        Attempt to solve the static response problem $(P^k)$ for ESL, or $(\tilde{P}^k)$ for F-ESL.
        Denote the found design $\hat{\mathbf{v}}^k$ and the associated Lagrange multipliers $\hat{\lambda}^k$.
        Update the outer iterate $\mathbf{v}^{k+1} = \hat{\mathbf{v}}^k$ and Lagrange multipliers $\lambda^{k+1} = \hat{\lambda}^k$.
        Update the iteration counter $k \leftarrow k+1$.
    end
    Let $(\hat{\mathbf{v}},\hat{\lambda}) = (\mathbf{v}^k,\lambda^k)$.
end
```

2.2. First-order ESL algorithm for static nonlinear response optimization

We propose to modify the equivalent static loads algorithm presented in Algorithm 1 by changing the loads in the linear response sub-problem from constant loads to design-dependent loads. The proposed first-order equivalent static loads algorithm is referred to as F-ESL. The modified sub-problem at outer iteration k is

$$\begin{aligned} \underset{\mathbf{v}\in\mathbb{R}^n}{\text{minimize}} \quad & f(\mathbf{v},\tilde{\mathbf{t}}(\mathbf{v};\mathbf{v}^k)) \\ \text{subject to} \quad & c_i(\mathbf{v},\tilde{\mathbf{t}}(\mathbf{v};\mathbf{v}^k)) \le 0 \qquad i=1,\dots,m \\ & \mathbf{v}^{\min} \le \mathbf{v} \le \mathbf{v}^{\max} \end{aligned} \tag{$\tilde{P}^k$}$$

The state vector $\tilde{\mathbf{t}}(\mathbf{v};\mathbf{v}^k)$ now solves the linear analysis equations with a design-dependent right hand side modelling a first-order approximation of the equivalent loads, i.e.

$$\begin{aligned} &\mathbf{K}(\mathbf{v})\,\tilde{\mathbf{t}}_\ell(\mathbf{v};\mathbf{v}^k) = \mathbf{f}_{\mathrm{eq},\ell}(\mathbf{v};\mathbf{v}^k) \quad \forall\,\ell \\ &\text{with } \mathbf{f}_{\mathrm{eq},\ell}(\mathbf{v};\mathbf{v}^k) \approx \mathbf{f}_{\mathrm{eq},\ell}(\mathbf{v}^k) + \nabla_{\mathbf{v}}\mathbf{f}_{\mathrm{eq},\ell}(\mathbf{v}^k)(\mathbf{v}-\mathbf{v}^k) \end{aligned} \tag{5}$$

The main advantage of this modification of the algorithm is that the sub-problem $(\tilde{P}^k)$ becomes a first-order approximation of the original problem (P) at the current iterate. This is a property that the sub-problem (P^k) used in the algorithm outlined in Algorithm 1 does not possess in general. This topic is explained in detail for linear dynamic response problems in [17] and exemplified in [18].

The sub-problem $(\tilde{P}^k)$ is a first-order approximation of the original problem, if at the kth outer iteration of F-ESL, the objective and constraint functions and their gradients are identical for the two problems, i.e. if

$$\begin{aligned} &f(\mathbf{v}^k,\mathbf{u}(\mathbf{v}^k)) = f(\mathbf{v}^k,\tilde{\mathbf{t}}(\mathbf{v}^k;\mathbf{v}^k)), \\ &c_i(\mathbf{v}^k,\mathbf{u}(\mathbf{v}^k)) = c_i(\mathbf{v}^k,\tilde{\mathbf{t}}(\mathbf{v}^k;\mathbf{v}^k)) \qquad \text{for all } i \end{aligned} \tag{6}$$

and

$$\begin{aligned} &\nabla_{\mathbf{v}} f(\mathbf{v}^k,\mathbf{u}(\mathbf{v}^k)) = \nabla_{\mathbf{v}} f(\mathbf{v}^k,\tilde{\mathbf{t}}(\mathbf{v}^k;\mathbf{v}^k)), \\ &\nabla_{\mathbf{v}} c_i(\mathbf{v}^k,\mathbf{u}(\mathbf{v}^k)) = \nabla_{\mathbf{v}} c_i(\mathbf{v}^k,\tilde{\mathbf{t}}(\mathbf{v}^k;\mathbf{v}^k)) \qquad \text{for all } i \end{aligned} \tag{7}$$

The objective and constraint functions in the nonlinear response problem (P) and linear response problem $(\tilde{P}^k)$ are essentially identical, with a difference only in the way the structural analysis is done. Hence, to show that the modified linear response problem $(\tilde{P}^k)$ is a first-order approximation of (P) at the current iterate it is sufficient to show that at $\mathbf{v} = \mathbf{v}^k$ the following relations are satisfied

$$\mathbf{u}(\mathbf{v}) = \tilde{\mathbf{t}}(\mathbf{v};\mathbf{v}^k) \text{ and } \frac{\partial \mathbf{u}(\mathbf{v})}{\partial v_i} = \frac{\partial \tilde{\mathbf{t}}(\mathbf{v};\mathbf{v}^k)}{\partial v_i} \text{ for all } i. \tag{8}$$

The first part is satisfied because of the definition of the equivalent static loads from Eq. (4). The second part requires some design sensitivity analysis. For simplicity, the sensitivity analysis is done for a single load case situation and the sub-index ℓ is therefore excluded.

The equivalent static loads $\mathbf{f}_{\mathrm{eq}}(\mathbf{v})$ for the design $\mathbf{v}$ are computed after the nonlinear response $\mathbf{u}(\mathbf{v})$ has been determined, and they are computed through

$$\mathbf{f}_{\mathrm{eq}}(\mathbf{v}) = \mathbf{K}(\mathbf{v})\mathbf{u}(\mathbf{v}) \tag{9}$$

This implies that the derivatives of the equivalent static loads are

$$\frac{\partial \mathbf{f}_{\mathrm{eq}}(\mathbf{v})}{\partial v_i} = \frac{\partial \mathbf{K}(\mathbf{v})}{\partial v_i}\mathbf{u}(\mathbf{v}) + \mathbf{K}(\mathbf{v})\frac{\partial \mathbf{u}(\mathbf{v})}{\partial v_i} \tag{10}$$

The design sensitivity analysis of the displacements in the linear response analysis sub-problem $(\tilde{P}^k)$ become

$$\begin{aligned} \frac{\partial \tilde{\mathbf{t}}(\mathbf{v};\mathbf{v}^k)}{\partial v_i} &= \frac{\partial \mathbf{K}^{-1}(\mathbf{v})}{\partial v_i}\left(\mathbf{f}^k_{\mathrm{eq}} + \nabla \mathbf{f}^k_{\mathrm{eq}}(\mathbf{v}-\mathbf{v}^k)\right) + \mathbf{K}^{-1}(\mathbf{v})\frac{\partial \mathbf{f}^k_{\mathrm{eq}}}{\partial v_i} \\ &= -\mathbf{K}^{-1}(\mathbf{v})\frac{\partial \mathbf{K}}{\partial v_i}\mathbf{K}^{-1}(\mathbf{v})\left(\mathbf{f}^k_{\mathrm{eq}} + \nabla \mathbf{f}^k_{\mathrm{eq}}(\mathbf{v}-\mathbf{v}^k)\right) + \mathbf{K}^{-1}(\mathbf{v})\frac{\partial \mathbf{f}^k_{\mathrm{eq}}}{\partial v_i} \end{aligned} \tag{11}$$

Inserting $\mathbf{v} = \mathbf{v}^k$ and then using the expression for the sensitivities of the equivalent loads from (10) results in

$$\begin{aligned} \frac{\partial \tilde{\mathbf{t}}(\mathbf{v}^k;\mathbf{v}^k)}{\partial v_i} &= -\mathbf{K}^{-1}(\mathbf{v}^k)\left(\frac{\partial \mathbf{K}(\mathbf{v}^k)}{\partial v_i}\mathbf{u}(\mathbf{v}^k) - \frac{\partial \mathbf{f}^k_{\mathrm{eq}}}{\partial v_i}\right) \\ &= -\mathbf{K}^{-1}(\mathbf{v}^k)\left(\frac{\partial \mathbf{K}(\mathbf{v}^k)}{\partial v_i}\mathbf{u}(\mathbf{v}^k) - \frac{\partial \mathbf{K}(\mathbf{v}^k)}{\partial v_i}\mathbf{u}(\mathbf{v}^k) - \mathbf{K}(\mathbf{v}^k)\frac{\partial \mathbf{u}(\mathbf{v}^k)}{\partial v_i}\right) \\ &= \frac{\partial \mathbf{u}(\mathbf{v}^k)}{\partial v_i} \end{aligned} \tag{12}$$

This establishes that sub-problem $(\tilde{P}^k)$ is a first-order approximation of (P) at $\mathbf{v}^k$.

2.3. Computational cost of the additional design sensitivity analysis

One argument for proposing the equivalent static loads algorithm is the possibility to avoid the design sensitivity analysis associated with nonlinear analysis. If we consider adding a first-order term in the approximation of the equivalent loads in the linear response sub-problem, then it is necessary to compute the derivatives of the equivalent loads with respect to the design variables.

Nonlinear static response analysis provides the displacements $\mathbf{u}$ by solving the system

$$\mathbf{r}(\mathbf{v}, \mathbf{u}(\mathbf{v})) = \mathbf{0} \tag{13}$$

The partial derivatives of the displacements be computed by solving the linear systems

$$\frac{\partial \mathbf{r}}{\partial \mathbf{u}} \frac{\partial \mathbf{u}}{\partial v_i} = -\frac{d\mathbf{r}}{dv_i} \tag{14}$$

with $\frac{\partial \mathbf{r}}{\partial \mathbf{u}}$ and $\frac{d\mathbf{r}}{dv_i}$ evaluated at $\mathbf{u}(\mathbf{v})$. Introduce the notation $\mathbf{K}_T(\mathbf{v}) = \frac{\partial \mathbf{r}}{\partial \mathbf{u}}$. Then the partial derivatives of the displacements are

$$\frac{\partial \mathbf{u}}{\partial v_i} = -\mathbf{K}_T^{-1}(\mathbf{v}) \frac{d\mathbf{r}}{dv_i} \tag{15}$$

Assume that the linear stiffness matrix $\mathbf{K}(\mathbf{v})$ is positive definite for all design variables in the feasible set. This is a common situation for e.g. sizing problems with positive lower bounds on the design variables. Consider the system $\mathbf{K}(\mathbf{v})\mathbf{t}(\mathbf{v}) = \mathbf{q}(\mathbf{v})$ with the design dependent load $\mathbf{q}(\mathbf{v}) = \mathbf{q}_0 + \sum_i v_i \mathbf{q}_i$ for some constant vectors $\mathbf{q}_0, \mathbf{q}_1, \ldots, \mathbf{q}_n$. Direct design sensitivity of the displacement vector $\mathbf{t}(\mathbf{v})$ gives

$$\frac{\partial \mathbf{t}(\mathbf{v})}{\partial v_i} = \frac{\partial \mathbf{K}^{-1}(\mathbf{v})}{\partial v_i} \mathbf{q}(\mathbf{v}) + \mathbf{K}^{-1}(\mathbf{v}) \frac{\partial \mathbf{q}(\mathbf{v})}{\partial v_i} = -\mathbf{K}^{-1} \frac{\partial \mathbf{K}}{\partial v_i} \mathbf{t} + \mathbf{K}^{-1}(\mathbf{v}) \mathbf{q}_i \tag{16}$$

Thus, the computation of the design sensitivity analysis amounts to solving a linear system with n right hand sides, i.e.

$$\nabla_{\mathbf{v}} \mathbf{t}(\mathbf{v}) = -\mathbf{K}^{-1}(\mathbf{v}) \mathbf{F}(\mathbf{v}, \mathbf{t}(\mathbf{v})) + \mathbf{K}^{-1}(\mathbf{v}) \mathbf{Q} \tag{17}$$

where

$$\mathbf{F}(\mathbf{v}, \mathbf{t}(\mathbf{v})) = \begin{pmatrix} \frac{\partial \mathbf{K}}{\partial v_1} \mathbf{t} & \cdots & \frac{\partial \mathbf{K}}{\partial v_n} \mathbf{t} \end{pmatrix} \text{ and } \mathbf{Q} = \begin{pmatrix} \mathbf{q}_1 & \cdots & \mathbf{q}_n \end{pmatrix} \tag{18}$$

It can be assumed that the stiffness matrix in this case is already assembled and factorized due to the preceding analysis required for solving the sub-problem. In this argumentation, we assume that the size of the problem and the number of degrees of freedom, are small enough such that an approach based on assembly and factorization of the relevant matrix is advantageous compared to other (e.g. iterative) approaches. The cost of computing and storing $\nabla_{\mathbf{v}} \mathbf{t}(\mathbf{v})$ in linear response optimization is therefore directly comparable to the cost of computing $\nabla_{\mathbf{v}} \mathbf{f}_{\text{eq}}(\mathbf{v})$. The gradient of the equivalent load is however done only *once per outer iteration* whereas the computation of design sensitivity analysis of the linear analysis displacements generally is done *once per inner iteration* and depending on the optimization algorithm used, possibly even more frequently.

3. Stabilizing the algorithm

We propose a set of additional modifications to the equivalent static loads algorithm to promote global convergence, i.e. convergence to a point satisfying first-order optimality conditions from any initial design. It is based on a trust-region approach [16]. This is just one of many possibilities. It is chosen because similar techniques are often used in structural optimization, they are easy to implement, and they generally work well in practice. We do however emphasize that the suggested approach is not sufficiently developed to ensure theoretical global convergence properties.

The first step in the modifications is to slightly reformulate the original problem by introducing artificial variables $y_i \geq 0$ and add a quadratic penalty term to the objective function. This approach is used in sequential convex optimization approaches in structural optimization (e.g. [21]). Instead of problem (P) we now consider the problem

$$\begin{array}{lll} \underset{\mathbf{v}\in\mathbb{R}^n, \mathbf{y}\in\mathbb{R}^m}{\text{minimize}} & f_N(\mathbf{v}, \mathbf{y}) := f(\mathbf{v}, \mathbf{u}(\mathbf{v})) + \frac{p}{2} \sum_i y_i^2 & \\ \text{subject to} & c_i(\mathbf{v}, \mathbf{u}(\mathbf{v})) - a_i y_i \leq 0 & i = 1, \ldots, m \\ & \mathbf{v}^{\min} \leq \mathbf{v} \leq \mathbf{v}^{\max} & \\ & \mathbf{y} \geq \mathbf{0} & \end{array} \tag{$P_{\mathbf{y}}$}$$

where $p \geq 0$ is a user-defined penalty parameter and $a_i \geq 0$ are user-defined constants. Compared to problem (P) infeasibility is now allowed at a (high) cost. If $a_i = 0$ for all i then the optimal artificial variables are all equal to zero and the problem reduces to the original problem (P). The a_i parameters are used to take care of the potentially varying scaling properties in the constraints.

Problem $(P_{\mathbf{y}})$ has a non-empty feasible set and under certain technical assumptions, notably smoothness of the objective and constraint functions and finite variable bounds. We are assured that the problem possesses optimal solutions, see e.g. [21]. The algorithm is initialized with a point $(\mathbf{v}^0, \mathbf{y}^0)$ that is feasible to $(P_{\mathbf{y}})$ that is chosen in the following manner. Pick a design variables vector $\mathbf{v}^0$ satisfying the variable bounds and compute artificial variables according to

$$y_i^0 = \begin{cases} \frac{1}{a_i} \max\{0, c_i(\mathbf{v}^0, \mathbf{u}(\mathbf{v}^0))\} & \text{for all } i \text{ such that } a_i > 0 \\ 0 & \text{otherwise.} \end{cases} \tag{19}$$

The algorithm then generates a sequence of iterates $\{(\mathbf{v}^k, \mathbf{y}^k)\}$ that are all feasible to $(P_{\mathbf{y}})$ by solving a sequence of sub-problems.

We then add a trust-region approach inspired by the approaches used for e.g. Sequential Quadratic Programming (SQP), see e.g. [16]. The constraint $\|\mathbf{v} - \mathbf{v}^k\|_\infty \leq \Delta_k$, where $\Delta_k > 0$ is the trust-region radius, is added to each of the sub-problems. The choice of norm is motivated by the ease of implementation of the trust-region constraints and the similarity to move limit strategies often used in structural optimization. They can be modelled by bound constraints on the design variables. Other commonly used norms in trust-region approaches, notably the Euclidean norm, could also be used. The sub-problem in the equivalent static loads algorithm retains the artificial variables and the modified (nonlinear) constraints and reads

$$\begin{array}{lll} \underset{\mathbf{v}\in\mathbb{R}^n}{\text{minimize}} & f_L^k(\mathbf{v}, \mathbf{y}) := f(\mathbf{v}, \tilde{\mathbf{t}}(\mathbf{v}; \mathbf{v}^k)) + \frac{p}{2} \sum_i y_i^2 & \\ \text{subject to} & c_i(\mathbf{v}, \tilde{\mathbf{t}}(\mathbf{v}; \mathbf{v}^k)) - a_i y_i \leq 0 & i = 1, \ldots, m \\ & \mathbf{v}^{\min} \leq \mathbf{v} \leq \mathbf{v}^{\max} & \\ & \|\mathbf{v} - \mathbf{v}^k\|_\infty \leq \Delta_k & \end{array} \tag{$\tilde{P}_{\mathbf{y}}^k$}$$

We let $(\tilde{\mathbf{v}}^k, \tilde{\mathbf{y}}^k)$ denote an optimal solution, i.e. a point satisfying the first-order optimality conditions, to the sub-problem $(\tilde{P}_{\mathbf{y}}^k)$. This point is generally not feasible to the original problem $(P_{\mathbf{y}})$ but can easily be modified to become feasible. The point $(\tilde{\mathbf{v}}^k, \bar{\mathbf{y}}^k)$ where

$$\bar{y}_i^k = \max\{\tilde{y}_i^k, \frac{1}{a_i} c_i(\tilde{\mathbf{v}}^k, \mathbf{u}(\tilde{\mathbf{v}}^k))\} \text{ for all } i \tag{20}$$

is feasible to $(P_{\mathbf{y}})$. By construction, this point will also be feasible to the next sub-problem if the design variables are updated.

The trust region radius is decreased when the estimated decrease and the actual change in the objective function are not well-aligned and it is allowed to increase when they are. We introduce some additional parameters $0 < \gamma_0 < \gamma_1 \leq 1 < \gamma_2$ and $0 < \eta_1 \leq \eta_2 \leq 1$. The η parameters are used to determine the quality of the sub-problem approximations and the γ parameters are used for the updates of the trust-region radius. The quality of the approximations are measured by

$$\rho_k = \frac{f_N(\mathbf{v}^k, \mathbf{y}^k) - f_N(\tilde{\mathbf{v}}^k, \bar{\mathbf{y}}^k)}{f_L^k(\mathbf{v}^k, \mathbf{y}^k) - f_L^k(\tilde{\mathbf{v}}^k, \tilde{\mathbf{y}}^k)} \tag{21}$$

The denominator is non-negative since $(\mathbf{v}^k, \mathbf{y}^k)$ is feasible (but not optimal in general) to the sub-problem whereas $(\tilde{\mathbf{v}}^k, \tilde{\mathbf{y}}^k)$ is optimal. The nominator measures the actual change in the objective function of the nonlinear response problem. The trust-region radius Δ_k is updated according to

$$\Delta_{k+1} = \begin{cases} \left[\gamma_0 \Delta_k, \gamma_1 \Delta_k\right] & \text{if } \rho_k < \eta_1 \\ \left[\gamma_1 \Delta_k, \Delta_k\right] & \text{if } \rho_k \in \left[\eta_1, \eta_2\right) \\ \left[\Delta_k, \gamma_2 \Delta_k\right] & \text{if } \rho_k \geq \eta_2 \end{cases} \tag{22}$$

This update rule means that if the objective function of the nonlinear response problem increases between outer iterations, then the trust-region is decreased. For a sub-problem with first-order approximation properties it can be expected that for some (sufficiently small) trust-region, a descent in the objective function is achieved.

If $\rho_k < \eta_1$ then the current iterate is kept and the trust-region radius is reduced and the sub-problem is resolved, i.e. $(\mathbf{v}^{k+1}, \mathbf{y}^{k+1}) \leftarrow (\mathbf{v}^k, \mathbf{y}^k)$. If on the other hand, $\rho_k \geq \eta_1$ then the trust-region radius is modified and the variables are updated according to $(\mathbf{v}^{k+1}, \mathbf{y}^{k+1}) \leftarrow (\tilde{\mathbf{v}}^k, \bar{\mathbf{y}}^k)$.

4. Application providing essentially convex sub-problems

The class of problems that we consider in this section are truss sizing problems with an objective function representing a measure similar to the compliance. We include an upper limit on the structural volume as the main constraint. The design variables represent the cross-section areas of the members of the truss ground structure. The class of minimum compliance problems is exceptionally well studied for linear static structural analysis [29]. In this situation it is possible to formulate the problems as convex in a number of different ways. Common formulations include semidefinite programs (SDP, e.g. [30,31], and [32]) and nonlinear optimization problems (e.g. [33]).

4.1. Worst-case minimum compliance problem

The way that the sub-problems are defined in the equivalent static load algorithm means that the constraints and objective function in principle remain the same. It is only the analysis assumptions and the external load that change when comparing the original problem and the sub-problems. This means that a compliance function written as $c(\mathbf{v}) = \mathbf{f}^T \mathbf{u}(\mathbf{v})$ remains unchanged in the sub-problem whereas the external force in the equilibrium equations is replaced by the equivalent load. For linear response analysis and stiffness matrices that are linear in the design variables, compliance functions can in general be reformulated in a convex way. However, when there is a mis-match between the loads defining the compliance and the external load the compliance function becomes a non-convex function. The (not so elegant) way around this obstacle is to modify the compliance function in the nonlinear response problem. Instead of writing it as $\mathbf{f}^T \mathbf{u}(\mathbf{v})$ we approximate it as $\mathbf{u}(\mathbf{v})^T \mathbf{K}(\mathbf{v}) \mathbf{u}(\mathbf{v})$. This way the force vector does not appear in the compliance function. The expense is that the compliance constraint is essentially the linear analysis compliance.

We therefore consider the worst-case minimum compliance problem

$$\begin{aligned} \underset{\mathbf{v} \in \mathbb{R}^n}{\text{minimize}} \quad & c_N(\mathbf{v}) = \max_{\ell} \left\{ \mathbf{u}_\ell^T(\mathbf{v}) \mathbf{K}(\mathbf{v}) \mathbf{u}_\ell(\mathbf{v}) \right\} \\ \text{subject to} \quad & \sum_{j=1}^{n} v_j l_j \leq V^{\max} \\ & \mathbf{v}^{\min} \leq \mathbf{v} \leq \mathbf{v}^{\max} \end{aligned} \tag{P_c}$$

where $V^{\max} > 0$ is a user provided upper limit on the structural volume and l_j is the (undeformed) length of the jth member in the truss ground structure. The feasible set in (P_c) is non-empty under relatively weak assumptions that are easily verified and typically satisfied in e.g. density based topology and sizing optimization. This situation can with advantage be used to study descent in a merit function without having to consider choice of penalty parameter. A suitable merit function is the objective function for the nonlinear response design problem (P_c).

The linear response sub-problem in the basic equivalent static load algorithm for this situation becomes

$$\begin{aligned} \underset{\mathbf{v} \in \mathbb{R}^n}{\text{minimize}} \quad & c_L(\mathbf{v}; \mathbf{v}^k) = \max_{\ell} \left\{ \mathbf{t}_\ell^T(\mathbf{v}; \mathbf{v}^k) \mathbf{K}(\mathbf{v}) \mathbf{t}_\ell(\mathbf{v}; \mathbf{v}^k) \right\} \\ \text{subject to} \quad & \sum_{j=1}^{n} v_j l_j \leq V^{\max} \\ & \mathbf{v}^{\min} \leq \mathbf{v} \leq \mathbf{v}^{\max} \end{aligned} \tag{P_c^k}$$

This problem can equivalently be reformulated as a problem in both design and state variables, i.e. a simultaneous analysis and design formulation,

$$\begin{aligned} \underset{\tau, \mathbf{v}, \mathbf{t}}{\text{minimize}} \quad & \tau \\ \text{subject to} \quad & \mathbf{f}_{\text{eq},\ell}(\mathbf{v}^k)^T \mathbf{t}_\ell \leq \tau & \forall \, \ell \\ & \mathbf{K}(\mathbf{v}) \mathbf{t}_\ell = \mathbf{f}_{\text{eq},\ell}(\mathbf{v}^k) & \forall \, \ell \\ & \sum_{j=1}^{n} v_j l_j \leq V^{\max} \\ & \mathbf{v}^{\min} \leq \mathbf{v} \leq \mathbf{v}^{\max}, \tau \geq 0 \end{aligned} \tag{S_c^k}$$

Problem (S_c^k) is equivalent to one of several possible convex problems, see e.g. [30,31], and [32]. One possibility is to resort to the SDP

$$\begin{aligned} \underset{\tau, \mathbf{v}}{\text{minimize}} \quad & \tau \\ \text{subject to} \quad & \begin{pmatrix} \tau & \mathbf{f}_{\text{eq},\ell}^T(\mathbf{v}^k) \\ \mathbf{f}_{\text{eq},\ell}(\mathbf{v}^k) & \mathbf{K}(\mathbf{v}) \end{pmatrix} \geq \mathbf{0} \quad \forall \, \ell \\ & \sum_{j=1}^{n} v_j l_j \leq V^{\max} \\ & \mathbf{v}^{\min} \leq \mathbf{v} \leq \mathbf{v}^{\max}, \tau \geq 0 \end{aligned} \tag{23}$$

Multiple equivalent convex formulations of problem (23) exist, see e.g. [34]. It is of course relevant for numerical efficiency which of them is used, but for the purpose of the relatively small-scale numerical experiments the SDP (23) is acceptable

The sub-problem in the modified version of the equivalent static loads algorithm for this particular situation becomes

$$\begin{aligned} \underset{\tau, \mathbf{v} \in \mathbb{R}^n}{\text{minimize}} \quad & \tau \\ \text{subject to} \quad & \begin{pmatrix} \tau & (\mathbf{f}_{\text{eq},\ell}^k + \nabla \mathbf{f}_{\text{eq},\ell}^k (\mathbf{v} - \mathbf{v}^k))^T \\ \mathbf{f}_{\text{eq},\ell}^k + \nabla \mathbf{f}_{\text{eq},\ell}^k (\mathbf{v} - \mathbf{v}^k) & \mathbf{K}(\mathbf{v}) \end{pmatrix} \geq \mathbf{0} \quad \forall \, \ell \\ & \sum_{j=1}^{n} v_j l_j \leq V^{\max} \\ & \mathbf{v}^{\min} \leq \mathbf{v} \leq \mathbf{v}^{\max} \end{aligned} \tag{24}$$

The matrix inequality in (24) is still linear in the design variables and the problem remains convex also after the modification.

The important sequences to consider are $\{c_{\text{N}}(\mathbf{v}^k)\}$ and $\{\mathbf{v}^k\}$. We know that $c_{\text{N}}(\mathbf{v}^k) = c_L(\mathbf{v}^k; \mathbf{v}^k)$ because of the definition of the equivalent loads evaluated at $\mathbf{v}^k$. We introduce the notation $c_L(\mathbf{v}; \mathbf{w})$ for the objective function in the linear problem where $\mathbf{w}$ is the design producing the equivalent loads and $\mathbf{v}$ is the design under consideration. If the algorithm in Algorithm 1 is started with a feasible $\mathbf{v}^0$ for problem (P_c) then every iterate will also be feasible. This follows since the feasible sets are identical and only involves the design variables and not the state variables between the nonlinear response problem and the linear response sub-problem. This implies that for a fixed iteration k that

$$c_{\text{N}}(\mathbf{v}^k) = c_L(\mathbf{v}^k; \mathbf{v}^k) \geq c_L(\tilde{\mathbf{v}}^k; \mathbf{v}^k) = c_L(\mathbf{v}^{k+1}; \mathbf{v}^k) \tag{25}$$

The first equality is due to the definition of the equivalent loads evaluated at $\mathbf{v}^k$ and the identical displacement fields for the two analysis situations. The first inequality is associated to the optimization analysis for the linear response sub-problem. The second equality is associated to the definition of the static loads. Then, the following question arises: what conditions are required to ensure that the inequality required for having an improvement in the objective function between two iterations, i.e. the inequality in

$$c_L(\mathbf{v}^{k+1}; \mathbf{v}^k) \overset{?}{\geq} c_L(\mathbf{v}^{k+1}; \mathbf{v}^{k+1}) = c_{\mathrm{N}}(\mathbf{v}^{k+1}; \mathbf{v}^{k+1}) \tag{26}$$

is satisfied? Even if the design is the same, the equivalent load and displacements vectors change because they are evaluated at a different point. There is no direct way for answering this question and for satisfying Eq. (26). Hence, at this point, there is (at least) one missing link to ensure progress in a merit function. In [1] this missing link is addressed through assumptions on the behaviour of the displacements between outer iterations (conditions 1–3 in [1]). In Section 6 we study the numerical convergence behaviour of ESL and F-ESL without any of these conditions.

For the worst-case minimum compliance problem (P_c) all the constraints are linear. If we assume that the starting point is feasible, all other iterates (both outer and inner iterates) remain feasible with a standard choice of optimization method for the sub-problems. For these problems, it is thus not necessary to include a penalty function or artificial variables. For the problem instances in the numerical experiments the penalty parameter $p = 0$ and the parameter for the artificial variable $a_1 = 0$. This implies that the artificial variable for the volume constraint can be chosen to be zero throughout. This kind of modelling allows us to study and understand if the reformulated sub-problems (23) or (24) provide descent in a merit function, for cases where merit and objective functions coincide.

The approximation quality indicator ρ_k in this situation simplifies to

$$\rho_k = \frac{c_N(\mathbf{v}^k) - c_N(\tilde{\mathbf{v}}^k)}{c_L(\mathbf{v}^k; \mathbf{v}^k) - c_L(\tilde{\mathbf{v}}^k; \mathbf{v}^k)} \tag{27}$$

If $\rho_k > 0$ then there is descent in the objective function. If, on the other hand, $\rho_k < 0$ then there is an increase in the objective function for the candidate design and the design is rejected, the trust-region radius is reduced, and a modified problem is solved. If the sub-problem is a good local approximation then it should be expected (due to continuity) that $\rho_k > 0$, for some trust-region radius $\Delta_k > 0$. We note that it is not necessary for the sub-problem to be a first-order approximation to provide a design that produces a decrease in the merit function, if the current outer iteration is sufficiently far away from an optimal design. But, if the approximation is not good enough, descent may not achieved and the algorithm may continue to decrease the trust-region radius until the termination criterion is met and the algorithm terminates.

5. Nonlinear structural modelling in detail

In this section we provide the details of two beam finite element formulations. The two implementations have been used in the numerical experiments to study the behaviour of the proposed first-order equivalent static method approach proposed. The details of some of the numerical experiments performed will be discussed in Section 6. The first formulation discussed considers a two-dimensional truss finite element based on Green's strain. The second formulation considers a two-dimensional co-rotational beam element based on Kirchhoff theory. In [26,27] a similar implementation for structural optimization of buckling-resistant trusses is considered.

5.1. Nonlinear truss modelling

The linear stiffness matrix for truss analysis can be written as

$$\mathbf{K}(\mathbf{v}) = \sum_j \frac{v_j E}{l_j^3} \mathbf{b}_j \mathbf{b}_j^T \tag{28}$$

where l_j is the undeformed length of the jth bar in the ground structure and $\mathbf{r}_j = \mathbf{b}_j / l_j$ contains the direction cosines of the jth bar. The nonlinear analysis strains are modelled as [35]

$$\epsilon_j(\mathbf{u}) = \frac{1}{l_j^2}(\mathbf{b}_j^T \mathbf{u}) + \frac{1}{2 l_j^2} \mathbf{u}^T \mathbf{B}_j \mathbf{u} \tag{29}$$

where the symmetric matrices $\mathbf{B_j} \in \mathbb{R}^{4\times4}$ in bar local coordinates are given by

$$\mathbf{B_j} = \begin{pmatrix} \mathbf{I} & -\mathbf{I} \\ -\mathbf{I} & \mathbf{I} \end{pmatrix} \tag{30}$$

where $\mathbf{I}$ is a 2 by 2 identity matrix. Introduce the following mapping in order to get a description of the potential energy

$$\mathbf{G}(\mathbf{v}, \mathbf{u}) = \sum_j \frac{v_j E}{l_j^3} \left((\mathbf{b}_j^T \mathbf{u}) \mathbf{B}_j + \frac{1}{4} (\mathbf{B}_j \mathbf{u})(\mathbf{B}_j \mathbf{u})^T \right) \tag{31}$$

The potential energy becomes

$$\Pi(\mathbf{v}, \mathbf{u}) = \frac{1}{2} \mathbf{u}^T \left(\mathbf{K}(\mathbf{v}) + \mathbf{G}(\mathbf{v}, \mathbf{u}) \right) \mathbf{u} - \mathbf{f}^T \mathbf{u} \tag{32}$$

The residual equations come from the necessary conditions for minimization of the potential energy over the state vectors $\mathbf{u}$, i.e. that

$$\mathbf{r}(\mathbf{v}, \mathbf{u}) = \nabla_{\mathbf{u}} \Pi(\mathbf{v}, \mathbf{u}) = \mathbf{0} \tag{33}$$

The residual equations are, in the numerical experiments, solved using a Newton–Raphson method with a backtracking line-search [16] to ensure sufficient decrease in the potential energy between iterations. The algorithm is terminated when $\|\mathbf{r}(\mathbf{v}, \mathbf{u})\|_\infty < \epsilon_a$ for some tolerance $\epsilon_a > 0$.

5.2. Nonlinear co-rotational beam modelling

In the following we present a short description of the beam finite element formulation considered. More details can be found in the textbook by M. A. Crisfield [35]. In particular, we consider a 2-D co-rotational beam element derived using Kirchhoff's theory, with the assumption of large displacements and rotations, and small strains. A schematic representation of an element in the initial and deformed configurations is given in Fig. 1.

The length of a beam element and its rotation in the global coordinates system are calculated as follows:

$$l = \sqrt{dx^2 + dy^2},\ l_d = \sqrt{Dx^2 + Dy^2},\ s = Dx/l_d,\ c = Dy/l_d,$$
$$\beta = \begin{cases} +\arcsin(s) & \text{if } (s \geq 0 \text{ and } c \geq 0) \text{ or } (s \leq 0 \text{ and } c \geq 0) \\ +\arccos(c) & \text{if } (s \geq 0 \text{ and } c \leq 0) \\ -\arccos(c) & \text{if } (s \leq 0 \text{ and } c \leq 0) \end{cases} \tag{34}$$

where dx, dy, Dx, Dy are the projected initial and final lengths

$$dx = x_2 - x_1,\ dy = y_2 - y_1,\ Dx = x_2 - x_1 + u_4 - u_1,\ Dy = y_2 - y_1 + u_5 - u_2. \tag{35}$$

Assuming an homogeneous cross section, the internal forces in the local coordinate system are:

$$\begin{bmatrix} \bar{N} \\ \bar{M}_1 \\ \bar{M}_2 \end{bmatrix} = \begin{bmatrix} \frac{EA}{l} & 0 & 0 \\ 0 & \frac{4EI}{l} & \frac{2EI}{l} \\ 0 & \frac{2EI}{l} & \frac{4EI}{l} \end{bmatrix} \begin{bmatrix} \bar{u} \\ \bar{\theta}_1 \\ \bar{\theta}_2 \end{bmatrix},\ \begin{bmatrix} \bar{u} \\ \bar{\theta}_1 \\ \bar{\theta}_2 \end{bmatrix} = \begin{bmatrix} l_d - l \\ u_3 - \phi \\ u_6 - \phi \end{bmatrix} \tag{36}$$

Eq. (36) in a compact form reads:

$$\bar{\mathbf{f}}^{\mathrm{int}} = \bar{\mathbf{D}}_e \bar{\mathbf{a}} \tag{37}$$

where $(\bar{\mathbf{f}}^{\mathrm{int}})^T = [\bar{N}\ \bar{M}_1\ \bar{M}_2]$, and $\bar{\mathbf{a}}^T = [\bar{u}\, \bar{\theta}_1\ \bar{\theta}_2]$. The vector of internal forces in the global coordinate system is calculated as follows:

$$\mathbf{f}^{\mathrm{int}} = \mathbf{B}^T \bar{\mathbf{f}}^{\mathrm{int}} \tag{38}$$

with

$$\mathbf{B} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} + \begin{bmatrix} \mathbf{v}^T \\ -\frac{1}{l_d} \mathbf{z}^T \\ -\frac{1}{l_d} \mathbf{z}^T \end{bmatrix} \tag{39}$$

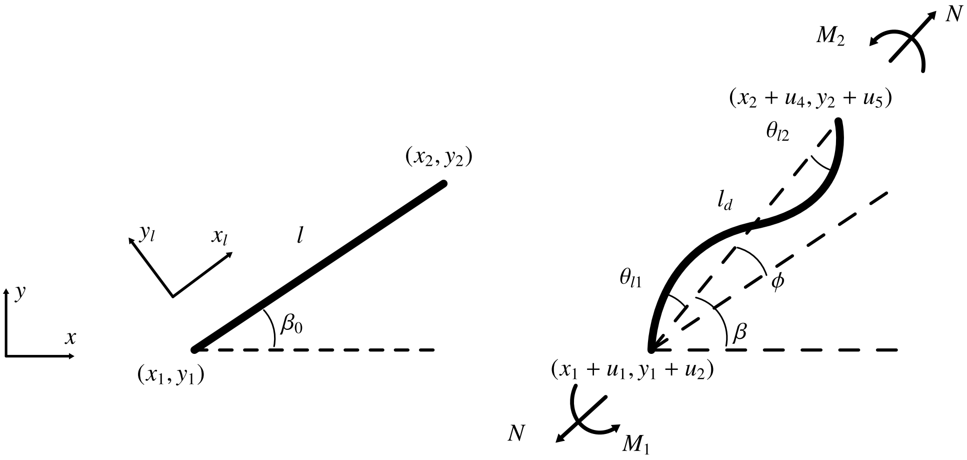


Fig. 1. Co-rotational beam element.

and

$$\begin{aligned} \mathbf{v}^T &= [-\cos(\beta)\ \ -\sin(\beta)\ 0\ \ \cos(\beta)\ \ \sin(\beta)\ 0], \\ \mathbf{z}^T &= [\sin(\beta)\ \ -\cos(\beta)\ 0\ \ -\sin(\beta)\ \ \cos(\beta)\ 0] \end{aligned} \tag{40}$$

The tangent stiffness matrix is

$$\mathbf{K}_T = \mathbf{B}^T\bar{\mathbf{D}}\mathbf{B} + \frac{\bar{N}}{l_d}\mathbf{z}\mathbf{z}^T + \frac{\bar{M}_1 + \bar{M}_2}{l_d^2}(\mathbf{v}\mathbf{z}^T + \mathbf{z}\mathbf{v}^T) \tag{41}$$

The nonlinear equilibrium equations are solved using an iterative Newton–Raphson algorithm with force control. The equilibrium equations read:

$$\mathbf{r}(\mathbf{v}, \mathbf{u}) = \mathbf{f}^{\text{int}}(\mathbf{v}, \mathbf{u}) - \mathbf{f}^{\text{ext}} \tag{42}$$

where $\mathbf{r}(\mathbf{v}, \mathbf{u})$ is the residual force vector; $\mathbf{f}^{\text{int}}(\mathbf{v}, \mathbf{u})$ is the internal force vector; $\mathbf{f}^{\text{ext}}$ is the external force vector. The iterative algorithm is terminated when $\|\mathbf{r}(\mathbf{v}, \mathbf{u})\|_\infty < \epsilon_a$ for some tolerance $\epsilon_a > 0$.

In the numerical examples of Section 6 we will use the two nonlinear beam formulations discussed in Section 5.1 and Section 5.2. The two formulations will be used to model and analyse truss structures, that is, structures made of beam elements connected by hinges and loaded at the joints. While the formulation of Section 5.1 considers true truss elements (i.e. elements loaded only axially and subjected only to axial deformations) the model of Section 5.2 considers slender beam elements with axial and bending stiffnesses and deformations. However, as the structures analysed will be characterized by beam elements with hinged connections and loaded only at the joints, in the case of the co-rotational beam element formulation of Section 5.2 the moments in the beams will be zero, as well as the end rotations. This means that, under these circumstances, in the numerical examples also the structures modelled with the co-rotational beam elements will actually behave as truss elements.

6. Numerical experiments

In the following section we present and discuss several numerical examples. These have been obtained considering different optimization problem formulations and structural design cases. For defining the optimization problem formulations, different combinations of objective functions (compliance, volume, selected displacements) and constraint functions (volume, displacements, stresses) are considered. Since the compliance and/or stress constraints are all based on compliance and stresses evaluated with linear analysis, it is acceptable to use an equivalent load that gives the same displacement field as for the nonlinear analysis. The compliance and stresses will follow.

The problems considered provide the smallest possible perturbation of a linear response structural optimization problem. This is already sufficient for illustrating and explaining the practical convergence properties of the ESL and F-ESL algorithms. The numerical implementation is based on choices which are suitable for the purposes of the numerical experiments, but that are not suitable in a production implementation for industrial and large-scale use of the algorithm. One example of such a implementation decision is that sub-problems which are equivalent to convex problems are solved twice: by a numerical optimization algorithm applied to the standard formulation, and by applying (a possibly different) numerical optimization algorithm to a convex reformulation of the problem.

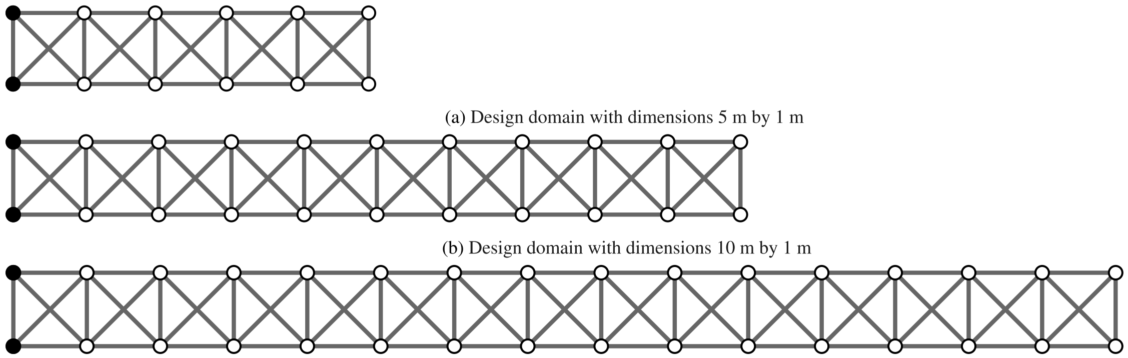
6.1. Computational considerations

The structural analysis and the routines for calling optimization solvers and computing objective functions and constraints are all implemented in Matlab. The nonlinear optimization problems are solved by the interior-point algorithm implemented in the `fmincon` function from the Matlab Optimization toolbox version 8.3. The SDPs (Section 4) are implemented in the modelling language CVX [36,37] and they are solved by the algorithm implemented in SeDuMi [38,39].

The tolerances used in the stopping criterion are presumably much smaller compared to the values used for the numerical experiments for the equivalent static loads algorithm in the literature. This is however a speculative statement since tolerances are rarely reported. The choice of tolerances allows for studies on the convergence behaviours of the equivalent static loads algorithm as the first-order optimality conditions are approached. The tolerances are chosen such that the optimality conditions are satisfied to within tolerances normally used for general purpose numerical optimization methods such as SQP and interior-point method (see e.g. [16]) for nonlinear optimization. This implies that the number of outer iterations increases, sometimes substantially, compared to the situation where the equivalent static loads algorithm would be terminated in a practical design situation.

The optimal design problems are scaled in several ways before being solved such that robust performance and high accuracy is achieved. The external loads are scaled by a factor between 10^{-6} and 10^{-4}, depending on the design case and the magnitude of the loads applied. The same scaling is applied to the modulus of elasticity for consistency. The design variables are normalized, such that their maximum value equals one. The scaling of the `fmincon` optimization solver is disabled. Finally, no efforts have been made to utilize the warm-start information that is present in the equivalent static loads algorithm for the sub-problems. This should not change the convergence behaviour under study, but it does affect the computation time.

The ESL and F-ESL algorithms are terminated when either one of the following is true: the outer iteration number k reaches the maximum


(a) Design domain with dimensions 5 m by 1 m

(b) Design domain with dimensions 10 m by 1 m

(c) Design domain with dimensions 15 m by 1 m

Fig. 2. Ground structures considered for the cantilever beam examples.

Table 1
Parameters and tolerances used in the implementation of ESL and F-ESL.

Notation	Description	Value
ϵ_a	Analysis tolerance	10^{-9}
ϵ_f	Feasibility tolerance	10^{-5}
ϵ_o	Optimality tolerance	10^{-5}
γ_0	Trust region radius decrease factor	0.1
γ_1	Trust region radius decrease factor	0.2
γ_2	Trust region radius increase factor	2.0
η_1	Trust region quality level	0.01
η_2	Trust region quality level	0.50

Table 2
Details of the ground structure for the cantilever beam examples considered in Section 6.3. N_x is the number of nodes in the horizontal x direction, N_y is the number of nodes in the vertical y direction, n is the number of structural elements, d is the number of degrees of freedom.

Domain	Dimensions	N_x	N_y	n	d
Cantilever	5 × 1	6	2	26	20
Cantilever	6 × 1	7	2	31	24
Cantilever	7 × 1	8	2	36	28
Cantilever	8 × 1	9	2	41	32
Cantilever	9 × 1	10	2	46	36
Cantilever	10 × 1	11	2	51	40
Cantilever	15 × 1	16	2	76	60

allowed value, set here to 200; the difference between subsequent design updates k and k+1 satisfies $\|\mathbf{x}_{k+1}-\mathbf{x}_k\|_\infty \leq \epsilon_o$. The entries of $\mathbf{x}$ are the normalized design variables, such that $\mathbf{v} = \bar{v}\,\mathbf{x}$, and $\bar{v}$ is a predefined value (e.g. 10^{-2} m^2). When the trust-region approach of Section 3 is used to stabilize ESL and F-ESL (leading to the nomenclature ESL + Stab and F-ESL + Stab), an additional convergence criterion is considered. This criterion terminates ESL + Stab or F-ESL + Stab at iteration k if the trust-region radius satisfies $\Delta_k \leq \epsilon_o$. The parameters and tolerances used in the implementation of the various algorithms are presented in Table 1, where ϵ_a is the tolerance adopted in the nonlinear analysis ($\|\mathbf{r}(\mathbf{v},\mathbf{u})\|_\infty < \epsilon_a$), and ϵ_f and ϵ_0 are, respectively, the feasibility and optimality tolerances adopted in fmincon. The values of γ_0, γ_1, γ_2, η_1, and η_2 were defined based on experience and initial numerical experiments, and led to a good behaviour of the algorithms in all numerical examples. These values are also similar to those suggested in [16]. Lastly, the material properties used throughout the numerical examples correspond to aluminium with $E = 70$ GPa.

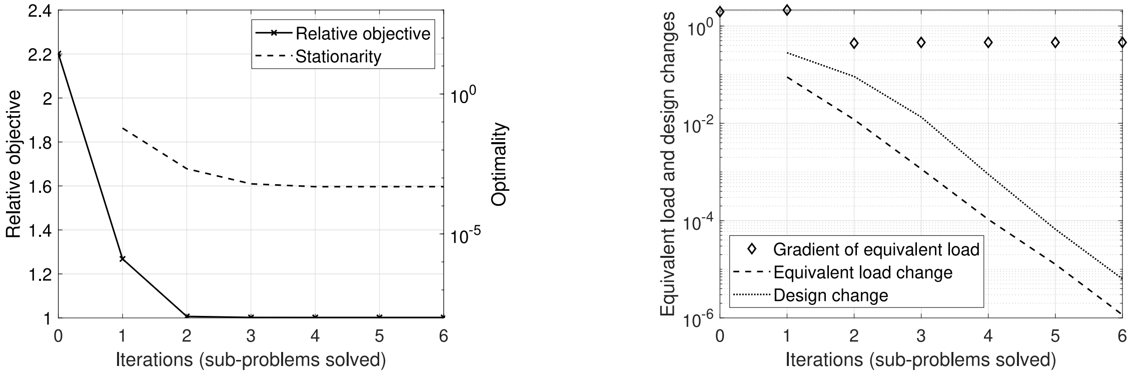
6.2. Experimental approach

The numerical experiments are all performed in the following manner. First the original problem, i.e. the problem based on nonlinear response analysis, is solved by a mathematical programming methods using analytical gradients for objective and constraint functions. The analytical gradients are computed either with adjoint or direct sensitivity analyses, depending on the case. Even if global optimality cannot be ensured due to non-convexity of the problem, this provides a reference design for later comparisons. Then the same problem is solved, this time with linear response analysis. This provides another design, and we can assess if it is reasonable to assume that the optimization based on nonlinear response analysis indeed provides a different optimal design.

Then the ESL and F-ESL algorithms (see Algorithm 1) with and without the stabilization approach described in Section 3 are employed on the same problem instances. At each outer iteration, corresponding to nonlinear analysis followed by computation of the equivalent loads, we monitor how close the obtained solution to the sub-problem is to satisfying the first-order optimality conditions (i.e. feasibility, stationarity, and complementarity) of the original problem. Also with ESL and F-ESL, all the gradient are calculated analytically. We also monitor other metrics that constitute the foundation for the theoretical results in [1]. In particular, we compute the norm of the derivatives of the equivalent static loads with respect to the design variables, i.e. $\|\nabla\mathbf{f}^k_{\text{eq}}\|_\infty$. For each outer iteration we report the difference in design $\|\mathbf{v}^{k+1} - \mathbf{v}^k\|_\infty$ between outer iterations, the norm of the difference in equivalent loads $\|\mathbf{f}^{k+1}_{\text{eq}} - \mathbf{f}^k_{\text{eq}}\|_\infty$, the norm of the Jacobian matrix of equivalent loads $\|\nabla_{\mathbf{v}}\mathbf{f}^k_{\text{eq}}\|_\infty$, the constraint violation $V(\mathbf{v}^k)$, and the current value of the objective function.

6.3. Minimum compliance problem with volume constraint — cantilever beam

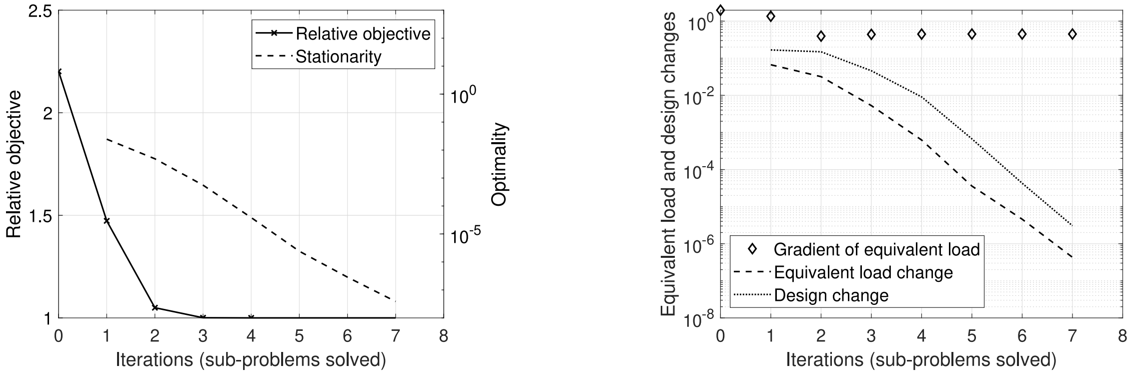
The minimum compliance problem instances are based on the design of a cantilever beam. The finite element modelling discussed in Section 5.1 is considered in this section. The dimensions of the design domains considered range from 5 m by 1 m, to 15 m by 1 m. In each case, the entire left side is fixed to a rigid wall. The boundary conditions and ground structures for the 5 m by 1 m, 10 m by 1 m, and 15 m by 1 cases are shown in Fig. 2. The details of the ground structures considered are given in Table 2, where N_x is the number of nodes in the horizontal x direction, N_y is the number of nodes in the vertical y direction, n is the number of structural elements, and d is the number of degrees of freedom. The ground structures do not contain any overlapping bars. A vertical point load is applied at the lower right corner of the design domain. The load corresponds to a mass of 10 metric tones. To avoid trivial design situations, the magnitude of the applied load has been selected to ensure that the response of the structures considered is effectively in the nonlinear range. In fact, only when the response of the structural system optimized goes in the nonlinear range, the accuracy of the particular equivalent static loads algorithm adopted plays a key role. The lower and upper bounds on the design variables are chosen as $v_i^{\min} = 10^{-4}$ m^2 and $v_i^{\max} = 10^{-2}$ m^2. In the initial design, all design are assigned the same value. This value



(a) Relative objective and stationarity

(b) Changes in design and equivalent static loads

Fig. 3. Convergence behaviour for ESL with trust-region stabilization when applied to the short 5 × 1 cantilever beam of Section 6.3.



(a) Relative objective and stationarity

(b) Changes in design and equivalent static loads

Fig. 4. Convergence behaviour for F-ESL with trust-region stabilization when applied to the short 5 × 1 cantilever beam of Section 6.3.

is chosen to ensure that the volume constraint is satisfied with equality. The volume limit V is set to

$$V = 0.1 \sum_j l_j v_j^{\max} \tag{43}$$

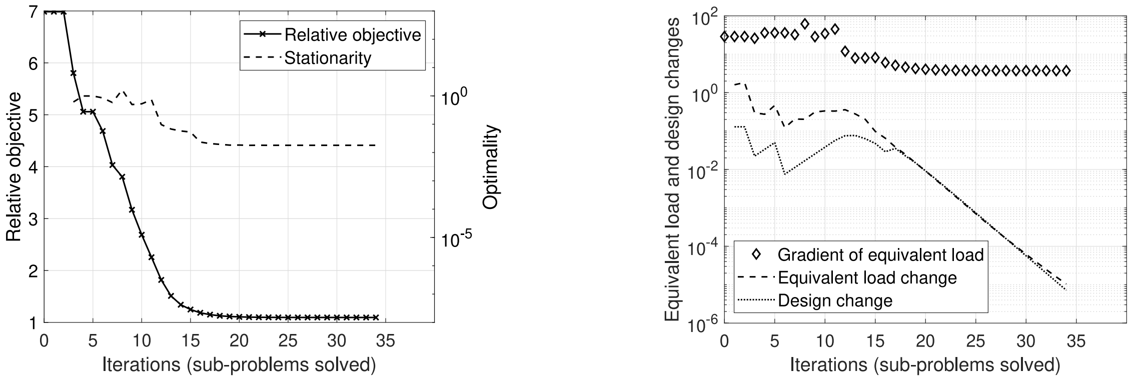
Fig. 3 shows the behaviour of the original equivalent static loads algorithm when applied to the minimum compliance problem (P_c) for the cantilever beam problem instance with an aspect ratio of 5:1. Fig. 4 shows the behaviour of the modified first-order equivalent static loads algorithm applied to the same problem instance, i.e. the algorithm with design dependent loads in the sub-problems. These figures provide several important observations which are also clearly displayed in all the other numerical results reported in this paper. The first observation is that the gradient of the equivalent static loads are not approaching zero as the design changes go to zero. This is otherwise one of the main assumptions for the theoretical results in [1]. The second observation is that the stationarity of the Lagrange function does not approach zero for the original ESL algorithm. It however does for the proposed F-ESL algorithm. This supports the theoretical results regarding the design sensitivity analysis presented above.

We also attempt to solve the same problem for aspect ratios of 10 × 1 and 15 × 1. For these problem instances, it is necessary to use the trust-region stabilization. The ground structure for these examples are shown in Fig. 2. Fig. 5 shows the behaviour of the basic equivalent static load algorithm, ESL, when applied to the 10 × 1 minimum compliance problem instance. Fig. 6 shows the behaviour of the proposed F-ESL algorithm applied to the same problem instance. Note that feasibility and complementarity measures are not presented in the tables and figures for the minimum compliance problems. This happens because the constraints are linear. Both feasibility and complementarity are satisfied to the requested tolerances for all iterates (both inner and outer).

Statistics from the original equivalent static load algorithm (ESL) and the proposed first-order algorithm (F-ESL) for the minimum compliance cantilever beam instances are presented in Table 3. The table reports the total number of sub-problems solved, the norm of the equivalent static loads vector at the final design, and the infinity norm of the first-order stationarity measure for the original problem. This measure is based on the final design and the Lagrange multipliers reported from the final sub-problem. The final column of the table lists the relative increase in the objective value compared to the reference design. If the algorithm failed to move from the initial design, the final three columns are marked with a dash, i.e. -.

For the problem instances with dimensions 10 × 1 m and 15 × 1 m the original ESL algorithm reaches the maximum number of outer iterations. The sequence of designs has increasing objective function values and the algorithm is terminated with a clearly non-optimal design. These instances are therefore not reported. The instance with dimensions 15 × 1 m also makes the proposed F-ESL algorithm stop after reaching the maximum number of outer iterations and fails to propose an optimal design.

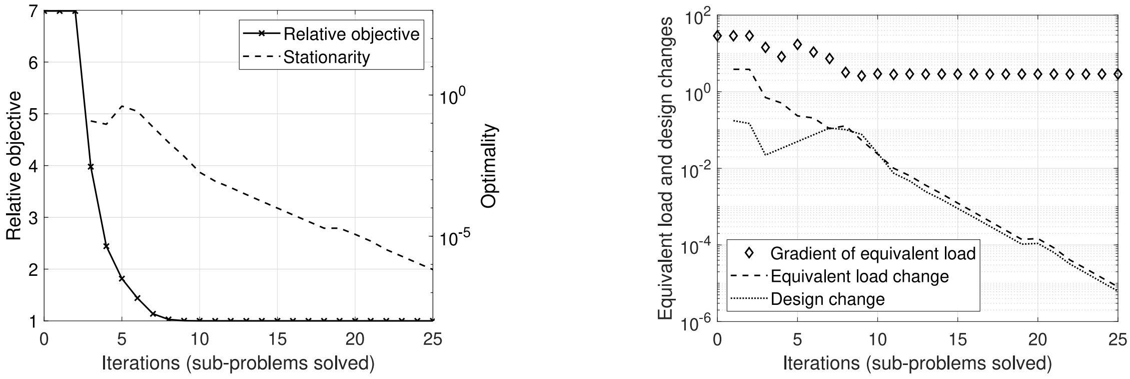
We note that the designs suggested by ESL do not satisfy the first-order optimality conditions for any of the three problem instances. Additionally, the found objective function values are higher than the reference design. For the short cantilever, the difference is small, the increase is only 0.2%, but it increases to 9.5% for the longer cantilever. F-ESL finds the same (to within the requested tolerances) optimal design as when applying an interior-point method to the original problem. Therefore, the relative objective difference is reported to be zero. For the problem instance with dimensions 15 × 1 m, the original equivalent



(a) Relative objective and stationarity

(b) Changes in design and equivalent static loads

Fig. 5. Convergence behaviour for ESL with trust-region stabilization when applied to the longer 10 × 1 cantilever beam of Section 6.3.



(a) Relative objective and stationarity

(b) Changes in design and equivalent static loads

Fig. 6. Convergence behaviour for F-ESL with trust-region stabilization when applied to the longer 10 × 1 cantilever beam of Section 6.3.

Table 3
Statistics when the original and modified equivalent static load method is applied to three cantilever beam problem instances of the minimum compliance problem with a volume constraint of Section 6.3.

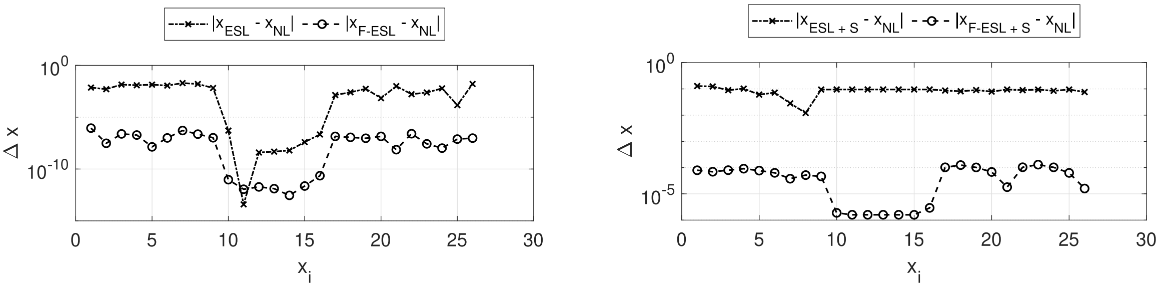
Domain	Algorithm	Outer itn.	$\|\nabla \mathbf{f}^k_{\mathrm{eq}}\|_\infty$	Stationarity	Objective diff.
5 × 1	ESL	6	0.46	$4.9 \cdot 10^{-4}$	+0.2%
6 × 1	ESL	8	0.73	$1.1 \cdot 10^{-3}$	+0.5%
7 × 1	ESL	9	1.08	$2.2 \cdot 10^{-3}$	+1.0%
8 × 1	ESL	12	1.52	$4.2 \cdot 10^{-3}$	+2.0%
9 × 1	ESL	17	2.15	$8.3 \cdot 10^{-3}$	+4.1%
5 × 1	ESL+Stab	6	0.46	$4.9 \cdot 10^{-4}$	+0.2%
6 × 1	ESL+Stab	11	0.73	$1.1 \cdot 10^{-3}$	+0.5%
7 × 1	ESL+Stab	9	1.08	$2.2 \cdot 10^{-3}$	+1.0%
8 × 1	ESL+Stab	12	1.52	$4.3 \cdot 10^{-3}$	+2.0%
9 × 1	ESL+Stab	20	2.15	$8.3 \cdot 10^{-3}$	+4.1%
10 × 1	ESL+Stab	34	3.74	$1.8 \cdot 10^{-2}$	+9.5%
15 × 1	ESL+Stab	8	–	–	–
5 × 1	F-ESL	7	0.45	$3.9 \cdot 10^{-8}$	+0.0%
6 × 1	F-ESL	10	0.70	$3.2 \cdot 10^{-8}$	+0.0%
7 × 1	F-ESL	11	1.04	$2.1 \cdot 10^{-7}$	+0.0%
8 × 1	F-ESL	13	1.52	$3.6 \cdot 10^{-7}$	+0.0%
9 × 1	F-ESL	15	2.14	$6.8 \cdot 10^{-7}$	+0.0%
10 × 1	F-ESL	22	2.89	$9.4 \cdot 10^{-7}$	+0.0%
5 × 1	F-ESL+Stab	7	0.45	$3.9 \cdot 10^{-8}$	+0.0%
6 × 1	F-ESL+Stab	10	0.70	$3.2 \cdot 10^{-8}$	+0.0%
7 × 1	F-ESL+Stab	11	1.04	$3.6 \cdot 10^{-7}$	+0.0%
8 × 1	F-ESL+Stab	13	1.52	$3.8 \cdot 10^{-7}$	+0.0%
9 × 1	F-ESL+Stab	19	2.14	$4.8 \cdot 10^{-7}$	+0.0%
10 × 1	F-ESL+Stab	25	2.89	$6.6 \cdot 10^{-7}$	+0.0%
15 × 1	F-ESL+Stab	54	13.63	$2.9 \cdot 10^{-5}$	+0.0%

static loads algorithm with stabilization fails to move away from the initial point. The trust-region quality measure for minimum compliance problems in (27) falls below zero for all sub-problems and the trust-region radius is thus reduced multiple times. This continues until the radius becomes too close to zero and the termination criteria is met without making any (outer) design updates. This illustrates the lack of approximation quality of the design-independent equivalent static loads used by ESL for this problem instance.

Table 3 shows that with both ESL and F-ESL the problem instances become more difficult to solve for increasing values of aspect ratio (i.e., with a more remarked gap between linear and nonlinear response) and number of design variables. With both algorithms, more outer iterations are required to meet the termination criteria. For problem instances with aspect ratios 10 × 1 and 15 × 1, ESL does not converge to final optimized designs. ESL with the trust-region approach, ESL+Stab, converges to final designs up to the case with aspect ratio 10 × 1, but for the last instance with aspect ratio 15 × 1 it also does not converge. Only F-ESL identifies optimized solutions that satisfy the stationarity measure for all problem instances considered.

6.4. Minimum volume with displacement constraint — clamped beam

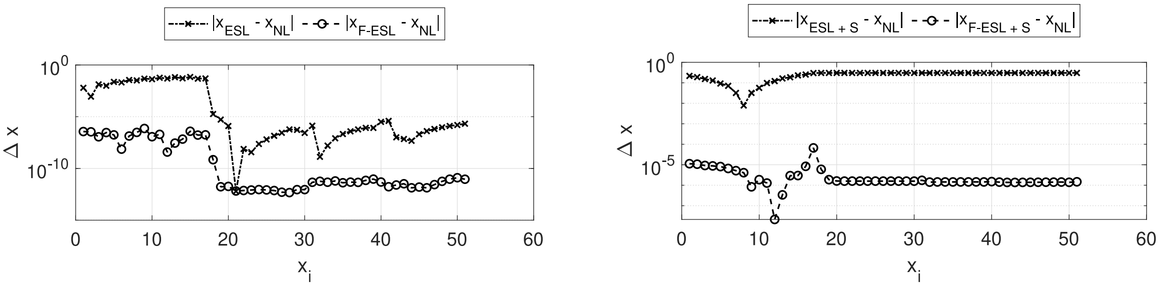
We now discuss sizing optimization of three truss structures modelled with the finite element formulation discussed in Section 5.2. The design variables are the cross sectional areas of the structural elements. The ground structures considered are shown in Fig. 2, and they refer to three clamped beams with different aspect ratios. As the aspect ratio increases, we expect the optimization problems to become more challenging for the ESL and F-ESL algorithms. In particular, the objective



(a) ESL vs F-ESL

(b) ESL + Stab vs F-ESL + Stab

Fig. 7. Comparison of optimized designs for the example of Section 6.4, 5 × 1 design case.



(a) ESL vs F-ESL

(b) ESL + Stab vs F-ESL + Stab

Fig. 8. Comparison of optimized designs for the example of Section 6.4, 10 × 1 design case.

function minimized is the structural volume V, with a constraint on the vertical displacement of the loaded node at the lower right corner, i.e. $\hat{u}$. Formally, the sizing optimization problem is stated as follows:

$$\begin{aligned} &\underset{\mathbf{x}\in\mathbb{R}^N}{\text{minimize}}\ V(\mathbf{v}) \\ &\text{subject to}\ \hat{u}(\mathbf{v}) - u^{\max} \le 0 \\ &\qquad v_i = x_i\,\bar{v},\ \text{for}\ i = 1,\dots,N \\ &\qquad x_i \in [0.1,\ 1],\ \text{for}\ i = 1,\dots,N \end{aligned} \tag{44}$$

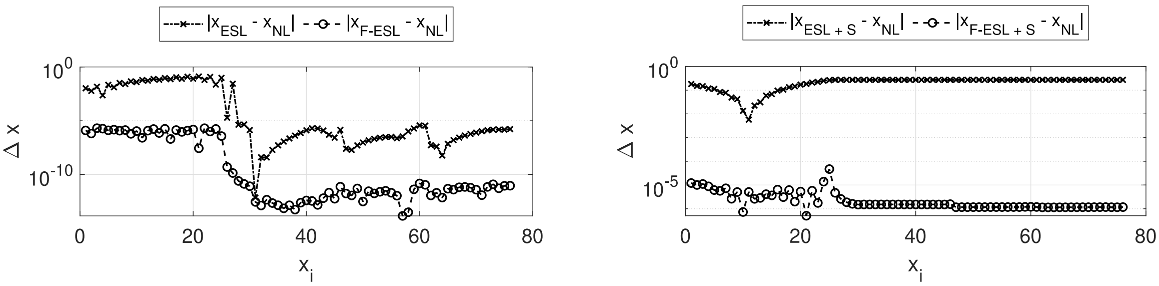
where $\bar{v} = 10^{-2}$ m^2 is the maximum cross sectional area allowed, and x_i and v_i are the entries of the vectors $\mathbf{x}$ and $\mathbf{v}$. In (44) the design variables (cross-sectional area) have a finite upper and lower bound, thus none of the elements in the ground structure is allowed to vanish. Problem (44) can be considered a sizing problem, as already mentioned. For computational purposes, the design variables are normalized, such that $v_i \in [10^{-3},\ 10^{-2}]$ for $i = 1,\dots,N$.

As mentioned above, we consider three cases, which depend on the dimensions of the design domain. The dimensions considered are 5 × 1 m, 10 × 1 m, and 15 × 1 m. The left side of the truss is fixed to a rigid wall. The ground structures do not contain any overlapping bars. A vertical point load is applied at the lower right corner of the design domain. The load corresponds to a mass of 30, 20, and 10 t for each design case, respectively. Additionally, an horizontal compressing load of 10 t is applied to each of the two nodes on the right-hand side. The maximum allowed vertical displacement of the lower right corner is 0.1, 0.5, and 1 m for each design case, respectively. In this example, the penalization parameters for the trust-region approach have been set as follows: $p = 10^3$, and $a = 10^{-3}$. All the design variables where initialized to 0.5. Problem (44) has been solved to local optimality in its original formulation, and approximately with ESL, F-ESL with and without the stabilization approach discussed in Section 3. For optimization, the interior-point algorithm of `fmincon` has been used in all of the optimization analyses. A maximum number of 200 iteration was allowed for ESL and F-ESL.

Table 4
Summary of the optimization results obtained with the basic (ESL) and first-order (F-ESL) algorithms, with and without stabilization, in the examples of Section 6.4.

Domain	Algorithm	Outer itn.	$\Vert\nabla\mathbf{f}^k_{\mathrm{eq}}\Vert_\infty$	Stationarity	Objective diff.
5 × 1	ESL	5	1.5e+02	2.8e−01	0.1%
10 × 1	ESL	11	1.0e+03	1.0e+00	1.1%
15 × 1	ESL	22	2.3e+03	2.2e+00	3.1%
5 × 1	ESL+Stab	200	1.4e+02	7.9e−01	23.1%
10 × 1	ESL+Stab	200	1.4e+03	4.2e−01	126.2%
15 × 1	ESL+Stab	25	2.8e+03	3.7e−01	124.2%
5 × 1	F-ESL	5	1.5e+02	8.2e−06	0.0%
10 × 1	F-ESL	7	1.0e+03	5.4e−06	0.0%
15 × 1	F-ESL	13	2.3e+03	3.1e−05	0.0%
5 × 1	F-ESL+Stab	9	1.5e+02	5.1e−06	0.0%
10 × 1	F-ESL+Stab	37	1.0e+03	1.7e−05	0.0%
15 × 1	F-ESL+Stab	82	2.3e+03	5.0e−06	0.0%

The results obtained in the optimization analyses are listed in Table 4. It can be observed that F-ESL always identifies design solutions with a final objective identical to that obtained by solving the original problem directly. Moreover, with F-ESL the final solutions numerically satisfy first-order optimality conditions. The solutions obtained with ESL are not stationary points, and differ significantly from the solution of the original problem. With the stabilization approach of Section 3, ESL does not converge for the 5 × 1 and 10 × 1 design cases within the given maximum number of iterations. For the case 15 × 1, ESL does not converge to a stationary point of the problem. The differences between the final solution obtained solving the original problem directly and the solutions obtained with ESL and F-ESL are plotted in Fig. 7, Fig. 8, and Fig. 9. It should be noted that $\mathbf{x}_{NL}$ refers to the solution obtained solving the original problem directly, $\mathbf{x}_{ESL+S}$ and $\mathbf{x}_{ESL}$ refer to the solution obtained with the ESL algorithm with and without stabilization, and similarly for the solutions obtained with the F-ESL algorithm $\mathbf{x}_{F\text{-}ESL+S}$ and $\mathbf{x}_{F\text{-}ESL}$. It can be observed that overall there is a clear tendency



(a) ESL vs F-ESL

(b) ESL + Stab vs F-ESL + Stab

Fig. 9. Comparison of optimized designs for the example of Section 6.4, 15 × 1 design case.

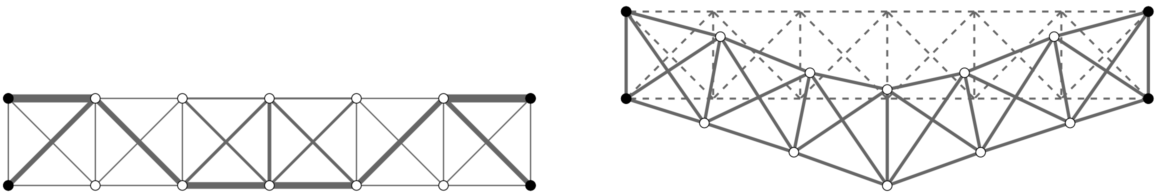
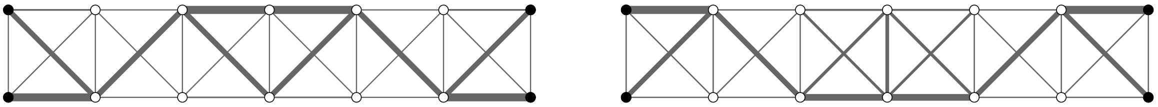


Fig. 10. Layout (10(a)) and normalized displaced configuration (10(b)) of the optimized structure obtained by solving problem (44) directly for the example of Section 6.5, 6 × 1 doubly clamped structure.



(a) ESL

(b) F-ESL

Fig. 11. Optimized structural layouts obtained for the example of Section 6.5 with ESL and F-ESL, 6 × 1 doubly clamped structure.

of the solutions obtained with F-ESL of being few orders of magnitude more accurate than the solutions obtained with ESL.

6.5. Minimum volume with displacement constraint — doubly clamped beam

We discuss a different design case for the same optimization problem (44) considered in Section 6.4, where the finite element formulation considered is also in this case the one presented in Section 5.2. In particular, we consider one design case, where the dimensions of the ground structured considered are 6x1 m. The ground structure is shown in Fig. 10(b) and identified by dashed lines. The left and right sides of the truss are fixed to a rigid wall. The ground structure does not contain any overlapping bars. A vertical point load is applied at the lower central joint. The load corresponds to a mass of 1500 t directed downwards. As in Section 6.3 here too the magnitude of the applied load has been defined in order to ensure that the response of the structures considered is effectively in the nonlinear range. In this way, we avoid trivial design scenarios and the accuracy of the particular equivalent static loads algorithms adopted plays a key role. The maximum allowed vertical displacement of the loaded joint is 0.3 m. All the remaining settings are kept as in the preceding example of Section 6.4.

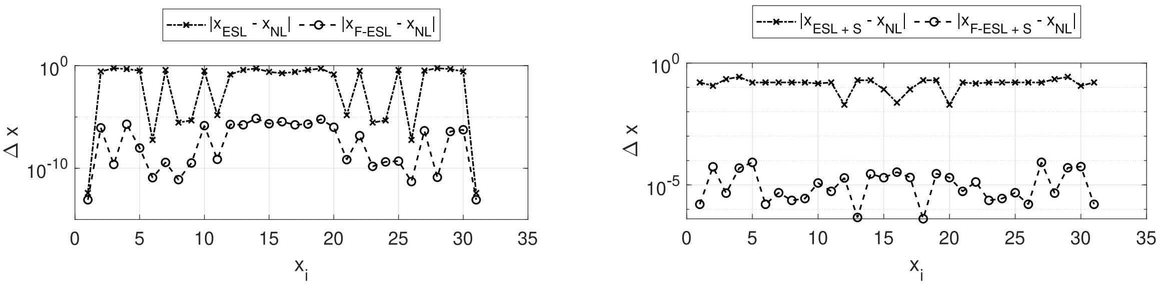
The results obtained in the optimization analyses are listed in Table 5. Also in this example, it can be observed that F-ESL identifies optimized design solutions with an objective function value identical to that obtained by solving the original problem directly. Moreover, this example confirms the ability of F-ESL to find design solutions that satisfy first-order optimality conditions. Fig. 10 shows the optimized layout and the associated displaced configuration obtained solving problem (44) directly with the interior-point algorithm of `fmincon`. For comparison, Fig. 11 shows the optimized layouts obtained with ESL and F-ESL. It can be observed that qualitatively only the design obtained with F-ESL is similar to that obtained by directly solving the problem at hand. This observation is confirmed by the graphs shown in Fig. 12, which show the differences between the final solution obtained by solving the problem at hand directly and the solutions obtained with ESL and F-ESL.

Table 5
Summary of the optimization results obtained with the basic (ESL) and first-order (F-ESL) algorithms, with and without trust-region stabilization, in the double clamped example of Section 6.5.

Domain	Algorithm	Outer itn.	$\|\nabla \mathbf{f}_{eq}^{k}\|_\infty$	Stationarity	Objective diff.
6 × 1	ESL	7	5.8e+03	1.3e+00	12.6%
6 × 1	ESL+Stab	15	5.0e+03	4.8e−01	23.8%
6 × 1	F-ESL	12	5.1e+03	4.2e−05	0.0%
6 × 1	F-ESL+Stab	13	5.1e+03	3.1e−05	0.0%

6.6. Minimum displacement with stress and volume constraints — clamped beam

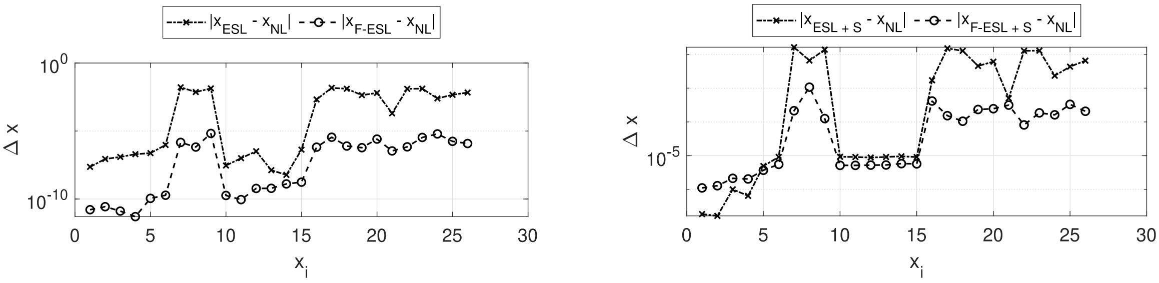
We consider a different optimization problem for the same structural design cases previously considered in Section 6.4, where the ground structures considered are shown in Fig. 2 and the finite element



(a) ESL vs F-ESL

(b) ESL + Stab vs F-ESL + Stab

Fig. 12. Comparison of optimized designs for the example of Section 6.5, 6 × 1 doubly clamped structure.



(a) ESL vs F-ESL

(b) ESL + Stab vs F-ESL + Stab

Fig. 13. Comparison of optimized designs for the example of Section 6.6, 5 × 1 design case.

formulation considered is presented in Section 5.2. We minimize a selected displacement with volume and stress constraints:

$$
\begin{aligned}
&\underset{\mathbf{x}\in\mathbb{R}^N}{\text{minimize}}\ \hat{u}(\mathbf{v})\\
&\text{subject to } V(\mathbf{v}) \leq V^{\max}\\
&\qquad\qquad \sigma_i(\mathbf{v}) \leq \sigma^{\max},\ \text{for } i = 1,\dots,N\\
&\qquad\qquad \sigma_i(\mathbf{v}) \geq \sigma^{\min},\ \text{for } i = 1,\dots,N\\
&\qquad\qquad v_i = x_i\,\bar{v},\ \text{for } i = 1,\dots,N\\
&\qquad\qquad x_i \in [0.1,\ 1],\ \text{for } i = 1,\dots,N
\end{aligned}
\tag{45}
$$

where x_i and v_i are the entries of the vectors $\mathbf{x}$ and $\mathbf{v}$. In (45) the minimized objective function is the vertical displacement of the loaded node in the lower right corner $\hat{u}$, V is the structural volume, $V^{\max}$ is the upper bound on the structural volume, σ_i is the axial stress in the ith bar, $\sigma^{\min}$ and $\sigma^{\max}$ are the lower and upper bounds of the allowed stresses, and lastly $\bar{v} = 10^{-2}$ m^2. The design variables (cross-sectional area) have finite upper and lower bounds. Thus, also problem (45) is a sizing problem. The design domain is defined by $v_i \in [10^{-3},\ 10^{-2}]$ for $i = 1,\dots,N$.

As in Section 6.4, here too we consider three cases, which depend on the aspect ratios of the design domains. The dimensions considered are 5 × 1 m, 10 × 1 m, and 15 × 1 m. A vertical point load is applied at the lower right corner of the design domain. The load corresponds to a mass of 50, 30, and 10 t for each design case, respectively. Additionally, an horizontal compressing load of 10 t is applied to each of the two nodes on the right-hand side. The maximum and minimum allowed axial stresses are $\sigma^{\min} = -300$ MPa, and $\sigma^{\max} = 300$ MPa. The volume fraction allowed is 50%. In this example, the penalization parameters for the trust-region approach are set as follows: $p = 10^3$, $a = 10^{-3}$. All the design variables where initialized to 0.95.

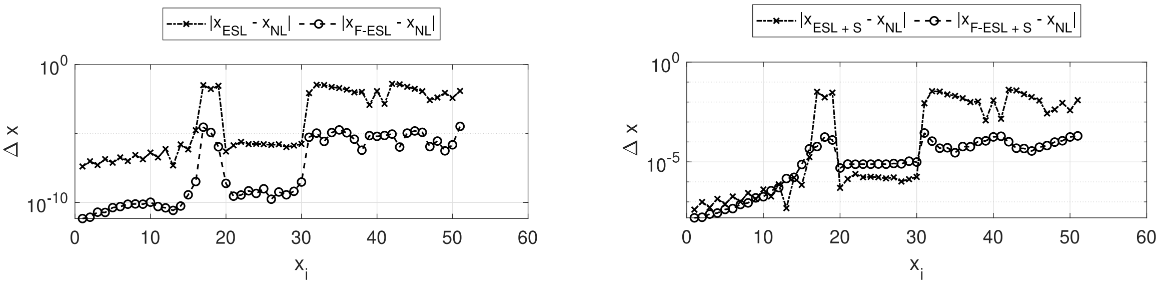
The results obtained in the optimization analyses are listed in Table 6. Also in this case, it can be observed that with F-ESL the final objective values are identical to that obtained by solving the original problem directly, and the final solutions are stationary points. The solutions obtained with ESL are not stationary points, and differ from the solution of the original problem. With the stabilization approach of Section 3, ESL does not converge for the 15 × 1 design case within the maximum number of iterations allowed. In none of the cases, ESL or ESL+Stab converge to stationary points with sufficient accuracy. The differences between the final solutions obtained solving the original problem directly and the solutions obtained with ESL and F-ESL are plotted in Fig. 13, Fig. 14, and Fig. 15. The tendency of F-ESL to identify final solutions few orders of magnitude more accurate than the solutions obtained with ESL is confirmed in these examples.

Table 6
Summary of the optimization results obtained with the basic (ESL) and first-order (F-ESL) algorithms, with and without trust-region stabilization, in the examples of Section 6.6.

Domain	Algorithm	Outer itn.	$\lVert\nabla \mathbf{f}^k_{\text{eq}}\rVert_\infty$	Stationarity	Objective diff.
5 × 1	ESL	4	8.7e+01	1.5e−02	0.01%
10 × 1	ESL	5	3.4e+02	1.5e−01	0.01%
15 × 1	ESL	200	2.0e+02	2.8e−01	0.01%
5 × 1	ESL+Stab	12	8.7e+01	1.5e−02	0.01%
10 × 1	ESL+Stab	5	3.4e+02	1.5e−01	0.01%
15 × 1	ESL+Stab	13	2.0e+02	2.8e−01	0.01%
5 × 1	F-ESL	5	8.8e+01	4.6e−06	0.00%
10 × 1	F-ESL	7	3.4e+02	2.2e−05	0.00%
15 × 1	F-ESL	7	2.0e+02	2.6e−05	0.00%
5 × 1	F-ESL+Stab	11	8.8e+01	1.2e−05	0.00%
10 × 1	F-ESL+Stab	15	3.4e+02	1.5e−05	0.00%
15 × 1	F-ESL+Stab	26	2.0e+02	2.3e−05	0.00%

7. Discussion of the numerical results

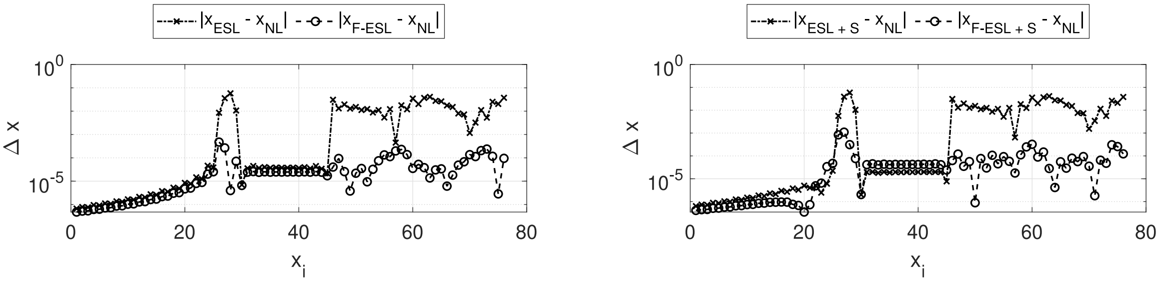
There are two main reasons for the behaviour of the basic ESL algorithm observed in the numerical examples of Section 6. These two can perhaps be seen as different sides of the same coin. First, the article [1] states that if the algorithm terminates in the sense of



(a) ESL vs F-ESL

(b) ESL + Stab vs F-ESL + Stab

Fig. 14. Comparison of optimized designs for the example of Section 6.6, 10 × 1 design case.



(a) ESL vs F-ESL

(b) ESL + Stab vs F-ESL + Stab

Fig. 15. Comparison of optimized designs for the example of Section 6.6, 15 × 1 design case.

the stopping conditions, then the design found by the last solved sub-problem with associated Lagrange multipliers satisfy the first-order optimality conditions for the original problem. A critical assumption in the proof of that statement in [1] is that the gradient of the equivalent loads is zero with respect to the design variables. The presented numerical results indicate that this conditions is not observed and that the first-order optimality conditions are not satisfied when the basic ESL algorithm terminates. Hence, there is a discrepancy between the assumptions underlying the theoretical results in [1] and the observed practical behaviour.

The second reason is that the equivalent loads in the basic ESL algorithm are only zeroth-order approximations, and as a consequence the loads do not change with the design variables in the sub-problems. This conscious decision is intended to avoid computing the design sensitivity analysis of the equivalent loads at each outer iteration. However, the consequence of this modelling decision is that the sub-problems are not local first-order approximations of the original problem at the current iterate. This explains the convergence issues that are observed in the numerical experiments. As a consequence, with ESL the trust-region radius goes to zero for some problem instances because there is a discrepancy between the predicted and actual changes in the objective functions that originates from the poor approximation of the original problem. Moreover, because of the poor approximation of the problem the first-order optimality conditions are not satisfied. This is analogous to using sequential convex programming algorithms with incorrect gradients of the objective and constraint functions.

In the case of the basic ESL algorithm, stabilization through the trust-region approach is unable to guide the algorithm to points satisfying first-order optimality conditions. The trust-region approach tends to accentuate the issues relating to the approximation in the sub-problem and it forces the algorithm to terminate without making any further design updates. This is an expected behaviour for the considered type of trust-region radius update schemes when the sub-problems do not approximate the original problem well enough. For the problem instances for which the proposed F-ESL algorithm finds designs satisfying first-order optimality conditions without stabilization, the trust-region approach can increase the number of outer iteration. However, it does not seem to hinder convergence to a good design. Again, this is expected, because stabilization approaches in optimization are intended to increase robustness potentially at the cost of decreased computational speed. The main advantage of the trust-region approach becomes visible for problem instances where F-ESL without stabilization fails to converge within the maximum number of outer iterations. This is exemplified in Table 3. For these problem instances, the trust-region is key for convergence.

One can argue that the problem statements and instances discussed herein are not well representing real-world applications, and hence that the numerical results do not represent the actual behaviour of ESL and F-ESL. Nevertheless, it is outside of the scope of this paper to investigate the behaviour for real-world or large-scale applications. We observe that none of the previously reported numerical studies, e.g. [1] or [3], presents any statistics on the norm of the gradients of the equivalent loads or quantifies how close the first-order optimality conditions are to being satisfied. Thus, the behaviour of the proposed F-ESL algorithm in large-scale and industry-relevant applications should be further studied in future work.

In general, one would like to solve a nonlinear response optimization problem directly. However, there may be cases where one does not have access to the nonlinear response finite element solver, as this is perhaps commercial software used as a black box. Thus, it is not possible to program the routines for the calculation of the analytical gradients necessary to perform gradient-based optimization. In these cases, one can resort to the F-ESL algorithm proposed in this article. Relying only on nonlinear response results combined with linear response optimization, one can still perform gradient-based optimization of the original nonlinear response optimization problem. It should be mentioned that a software implementation of the proposed F-ESL algorithm does not require access to a different type of information compared to the basic ESL algorithm. ESL couples a commercial software for nonlinear response analysis and a gradient-based optimization software for designing the associated linear system with equivalent static loads. F-ESL requires: the response of the nonlinear system; the Jacobian matrix of the residual of the nonlinear equilibrium equations, which is computed anyway for evaluating the response of the nonlinear system; and, as ESL, access to the response and sensitivity analysis routines for the optimization of the linear system in the sub-problems. Thus, F-ESL requires access to the same sources of information as ESL does and can thus be implemented within commercial structural analysis software under the same circumstances as the basic ESL algorithm.

8. Conclusions

In this paper, the basic equivalent static load (ESL) algorithm for optimal design of nonlinear static response problems is modified. The proposed modified equivalent static load algorithm (F-ESL) uses first-order approximation of the equivalent static loads in the static response sub-problems. The proposed F-ESL algorithm has, as a result, the first-order approximation properties that are missing in the original ESL algorithm. It is thus possible to guarantee that if F-ESL terminates and satisfies the proposed stopping conditions, then the found design satisfies the necessary first-order optimality conditions of the original optimization problem. Moreover, a trust-region approach is proposed in order to provide a more stable convergence of both ESL and F-ESL towards final optimized designs. The proposed F-ESL algorithm and the associated stabilization technique are assessed on a series of numerical examples. To facilitate the reproducibility of the results, the numerical examples consist of sizing optimization of truss structures with geometric nonlinearity. For the structural analysis, a nonlinear truss formulation and a nonlinear co-rotational beam formulation are considered. Different optimization problem formulations are considered for structures with different aspect ratios and boundary conditions. The numerical results confirm the capability of the proposed F-ESL algorithm in converging to designs that satisfy first-order optimality conditions, while requiring consistently few outer iterations across all the examples considered. F-ESL can thus be seen as an optimization approach for nonlinear static problems with first-order approximation properties.

Replication of results

The equivalent static loads algorithm(s), the optimization problems considered, and the structural analysis equations are described in sufficient detail to be implemented. Additionally, all parameters and tolerances used in the numerical experiments are presented in the paper. The implementations used for producing the numerical results follow the descriptions in the paper. For problems with convex sub-problems it is thus expected that the results can be reproduced to within the stated tolerances. Deviations can be expected if, for example, different analysis and/or optimization techniques are used. For the problems with non-convex sub-problems there is of course the risk that a different sequence of designs is found and the results can thus deviate in this situation.

Funding information

This research did not receive any specific grant from funding agencies in the public, commercial, or non-profit sectors.

CRediT authorship contribution statement

Mathias Stolpe: Conceptualization, Methodology, Formal analysis, Investigation, Writing – original draft. **Nicolò Pollini:** Conceptualization, Methodology, Formal analysis, Investigation, Writing – original draft.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgements

We thank three anonymous reviewers for their valuable comments that helped improve the quality of the article.

References

[1] Shin M-K, Park K-J, Park GJ. Optimization of structures with nonlinear behavior using equivalent loads. Comput Methods Appl Mech Engrg 2007;196(4–6):1154–67.
[2] Park GJ. Technical overview of the equivalent static loads method for non-linear static response structural optimization. Struct Multidiscip Optim 2011;43(3):319–37.
[3] Ahmad Z, Sultan T, Zoppi M, Abid M, Park GJ. Nonlinear response topology optimization using equivalent static loads—case studies. Eng Optim 2017;49(2):252–68.
[4] Lee HA, Park GJ. A software development framework for structural optimization considering non linear static responses. Struct Multidiscip Optim 2015;52(1):197–216.
[5] Kang BS, Choi WS, Park GJ. Structural optimization under equivalent static loads transformed from dynamic loads based on displacement. Comput Struct 2001;79(2):145–54.
[6] Choi WS, Park GJ. Structural optimization using equivalent static loads at all time intervals. Comput Methods Appl Mech Engrg 2002;191(19–20):2105–22.
[7] Park GJ, Kang BS. Validation of a structural optimization algorithm transforming dynamic loads into equivalent static loads. J Optim Theory Appl 2003;118(1):191–200.
[8] Park KJ, Lee JN, Park GJ. Structural shape optimization using equivalent static loads transformed from dynamic loads. Internat J Numer Methods Engrg 2005;63(4):589–602.
[9] Jeong S, Yi S, Kan C, Nagabhushana V, Park G. Structural optimization of an automobile roof structure using equivalent static loads. Proc Inst Mech Eng D 2008;222(11):1985–95.
[10] Jeong S, Yoon S, Xu S, Park G. Non-linear dynamic response structural optimization of an automobile frontal structure using equivalent static loads. Proc Inst Mech Eng D 2010;224(4):489–501.
[11] Hong EP, You BJ, Kim CH, Park GJ. Optimization of flexible components of multibody systems via equivalent static loads. Struct Multidiscip Optim 2010;40(1):549–62.
[12] Lee H-A, Park GJ. Nonlinear dynamic response topology optimization using the equivalent static loads method. Comput Methods Appl Mech Engrg 2015;283:956–70.
[13] Karev A, Harzheim L, Immel R, Erzgräber M. Comparison of different formulations of a front hood free sizing optimization problem using the ESL-method. In: World congress of structural and multidisciplinary optimisation. Springer; 2017, p. 933–51.
[14] Karev A, Harzheim L, Immel R, Erzgräber M. Free sizing optimization of a front hood using the ESL method: overcoming challenges and traps. Struct Multidiscip Optim 2019;60(4):1687–707.
[15] Choi W-H, Lee Y, Yoon J-M, Han Y-H, Park G-J. Structural optimization for roof crush test using an enforced displacement method. Int J Automot Technol 2018;19(2):291–9.
[16] Nocedal J, Wright SJ. Numerical Optimization. Springer New York; 2006,
[17] Stolpe M. On the equivalent static loads approach for dynamic response structural optimization. Struct Multidiscip Optim 2014;50(6):921–6.
[18] Stolpe M, Verbart A, Rojas-Labanda S. The equivalent static loads method for structural optimization does not in general generate optimal designs. Struct Multidiscip Optim 2018;58(1):139–54.
[19] Triller J, Immel R, Timmer A, Harzheim L. The difference-based equivalent static load method: an improvement of the ESL method's nonlinear approximation quality. Struct Multidiscip Optim 2021;63(6):2705–20.
[20] Triller J, Immel R, Harzheim L. Difference-based equivalent static load method with adaptive time selection and local stiffness adaption. Struct Multidiscip Optim 2022;65(3):1–17.
[21] Svanberg K. A class of globally convergent optimization methods based on conservative convex separable approximations. SIAM J Optim 2002;12(2):555–73.

[22] Saka M. Optimum design of nonlinear space trusses. Comput Struct 1988;30(3):545–51.

[23] Pedersen CB. Topology optimization of 2D-frame structures with path-dependent response. Internat J Numer Methods Engrg 2003;57(10):1471–501.

[24] Kaveh A, Rahami H. Nonlinear analysis and optimal design of structures via force method and genetic algorithm. Comput Struct 2006;84(12):770–8.

[25] Kaveh A, Rezaei M. Optimum topology design of geometrically nonlinear suspended domes using ECBO. Struct Eng Mech Int J 2015;56(4):667–94.

[26] Madah H, Amir O. Truss optimization with buckling considerations using geometrically nonlinear beam modeling. Comput Struct 2017;192:233–47.

[27] Madah H, Amir O. Concurrent structural optimization of buckling-resistant trusses and their initial imperfections. Int J Solids Struct 2019;162:244–58.

[28] Park GJ, Lee Y. Discussion on the optimality condition of the equivalent static loads method for linear dynamic response structural optimization. Struct Multidiscip Optim 2019;59(1):311–6.

[29] Bendsøe M, Sigmund O. Topology Optimization - Theory, Methods, and Applications. Springer Verlag; 2003.

[30] Ben-Tal A, Nemirovski A. Robust truss topology design via semidefinite programming. SIAM J Optim 1997;7(4):991–1016.

[31] Ben-Tal A, Nemirovski A. Lectures on Modern Convex Optimization: Analysis, Algorithms, and Engineering Applications. SIAM; 2001, p. 488.

[32] Achtziger W, Kočvara M. Structural Topology optimization With eigenvalues. SIAM J Optim 2007;18(4):1129–64.

[33] Achtziger W, Bendsøe M, Ben-Tal A, Zowe J. Equivalent displacement based formulations for maximum strength truss topology design. Impact Comput Sci Eng 1992;4(4):315–45.

[34] Kočvara M. Decomposition of arrow type positive semidefinite matrices with application to topology optimization. Math Program 2021;190(1–2):105–34.

[35] Crisfield MA. Nonlinear Finite Element Analysis of Solids and Structures. Volume 1: Essentials. John Wiley & Sons; 2000.

[36] Grant M, Boyd S. Graph implementations for nonsmooth convex programs. In: Blondel V, Boyd S, Kimura H, editors. Recent Advances in Learning and Control. Lecture Notes in Control and Information Sciences, Springer-Verlag Limited; 2008, p. 95–110.

[37] Grant M, Boyd S. CVX: Matlab software for disciplined convex programming, version 2.1, . 2014,.

[38] Sturm JF. Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. Optim Methods Softw 1999;11–2(1–4):625–53.

[39] Sturm JF. Implementation of interior point methods for mixed semidefinite and second order cone optimization problems. Optim Methods Softw 2002;17(6):1105–54.