SIAM J. OPTIM.
Vol. 26, No. 4, pp. 2314–2354

A SUBGRADIENT METHOD FOR FREE MATERIAL DESIGN*

MICHAL KOČVARA†, YURII NESTEROV‡, AND YU XIA§

Abstract. A small improvement in the structure of a material could potentially lower manufacturing costs. Thus, the free material design can be formulated as an optimization problem. However, due to its large scale, second-order methods cannot solve the free material design problem in a reasonable time. We formulate the free material optimization (FMO) problem into a saddle-point form in which the inverse of the stiffness matrix $A(E)$ in the constraint is eliminated. The size of $A(E)$ is generally large, denoted as $N \times N$. This is the first formulation of FMO without $A(E)^{-1}$. We apply the primal-dual subgradient method [Y. Nesterov, *Math. Program.,* 120 (2009), pp. 221–259] to solve the restricted saddle-point formula. This is the first gradient-type method for FMO. Each iteration of our algorithm takes a total of $\mathcal{O}(N^2)$ floating-point operations and an auxiliary vector storage of size $\mathcal{O}(N)$, compared with formulations having the inverse of $A(E)$ which requires $\mathcal{O}(N^3)$ arithmetic operations and an auxiliary vector storage of size $\mathcal{O}(N^2)$. To solve the problem, we developed a closed-form solution to a semidefinite least squares problem and an efficient parameter update scheme for the gradient method. We also approximate a solution to the bounded Lagrangian dual problem. The problem is decomposed into small problems, each having only an unknown of $k \times k$ ($k = 3$ or 6) matrix, and can be solved in parallel. The iteration bound of our algorithm is optimal for a general subgradient scheme. Finally, we present promising numerical results.

Key words. fast gradient method, Nesterov's primal-dual subgradient method, free material optimization, large-scale problems, first-order method, saddle-point, Lagrangian, complexity, duality, constrained least squares

AMS subject classifications. 90C90, 90C06, 90C25, 90C30, 90C47, 90-08

DOI. 10.1137/15M1019660

1. Introduction. The approach of *free material optimization (FMO)* is to optimize the material structure, while the distribution of material and the material itself can be freely varied. For example, FMO has been used to improve the overall material arrangement in air frame design (www.plato-n.org). The fundamentals of FMO were introduced in [3, 19]. The model was further developed in [2, 24] and elsewhere. In the model, the elastic body of the material under consideration is represented as a bounded domain with a Lipschitzian boundary in a two- or three-dimensional Euclidean space, depending on the design requirement. For computational purpose, the domain is discretized into m finite elements, $\Omega = (\Omega_1, \ldots, \Omega_m)$, so that all of the points in the same element are considered to have the same property.

Let $u(x)$ denote the *displacement vector* of the body at point x under load. Denote

*Received by the editors May 4, 2015; accepted for publication (in revised form) June 1, 2016; published electronically November 1, 2016.
http://www.siam.org/journals/siopt/26-4/M101966.html
Funding: The research of the third author was supported in part by a discovery grant of NSERC.
†School of Mathematics, The University of Birmingham, Birmingham B15 2TT, UK, and Institute of Information Theory and Automation, Academy of Sciences of the Czech Republic, 117 20 Prague 1, Czech Republic (m.kocvara@bham.ac.uk).
‡Center of Operations Research and Econometrics Catholic University of Louvain, B-1348 Louvain-la-Neue, Belgium, and National Research University Higher School of Economics, Moscow 101000, Russia (yurii.nesterov@uclouvain.be).
§Faculty of Business Administration, Lakehead University, Thunder Bay, ON P7B 5E1, Canada (yxia@lakeheadu.ca).

the *(small-)strain tensor* as

$$e_{ij}\left(u(x)\right) \stackrel{\text{def}}{=} \frac{1}{2}\left(\frac{\partial u(x)_i}{\partial x_j} + \frac{\partial u(x)_j}{\partial x_i}\right).$$

Let $\sigma_{ij}(x)$ $(i, j = 1, \ldots, 3)$ denote the *stress tensor*. The system is assumed to follow the Hooke's law—the stress is a linear function of the strain,

$$\sigma_{ij}(x) = E_{ijkl}(x) e_{kl}\left(u(x)\right) \quad \text{(in tensor notation)},$$

where E is the *(plain-stress) elasticity tensor* of order 4, which maps the strain to the stress tensor. The matrix E measures the degree of deformation of a material under external loads, and it is a symmetric positive semidefinite matrix of order 3 for the two-dimensional and of order 6 for the three-dimensional material design problem. The diagonal elements of $E(x)$ measure the stiffness of the material at x in the coordinate directions. Hence, the trace of E is used to measure the cost (resource used) of a material in the model.

Denote by I_k the identity matrix of order k and by S_k^{+m} the direct product of m cones of symmetric positive semidefinite $k \times k$-matrices,

$$S_k^{+m} \;=\; \underbrace{S_k^+ \times \cdots \times S_k^+}_{m \text{ times}}.$$

For a $k \times k$ symmetric matrix M, let $M \succeq 0$ denote $M \in S_k^+$.

Let E_i denote the elasticity tensor of order 4 for the ith element Ω_i: The E_i's are considered to be constant on each Ω_i but can be different for different Ω_i's, and they are the design variables of the FMO model,

$$E \;=\; (E_1, \ldots, E_m), \qquad E_i \;\succeq\; 0,\; i = 1, \ldots, m.$$

The design problem is to find a structure that is low "cost" (the tensor E having small trace) and is stable under given multiple independent loads (forces). There are different formulas of the FMO problem that depend on the design needs. This paper focuses on the *minimum-cost FMO* problem, which is to design a material structure that can withstand a given set of loads in the worst-case scenario while minimizing the trace of E. Below we describe the model based on [11].

The "cost"—stiffness of the material—is measured by the trace of E: $\operatorname{tr}(E) = \sum_{i=1}^m \operatorname{tr}(E_i) = \langle I_k, E\rangle$. For each $i \in \{1, \ldots, m\}$, $\operatorname{tr}(E_i)$ is lower bounded to avoid singularity in the FMO model. The constraints for the pointwise stiffness upper and lower bounds are

$$\operatorname{tr} E_i \le \rho_u^{(i)}, \qquad \operatorname{tr} E \ge \rho_L^{(i)}.$$

From the engineering literature, we know that the dynamic stiffness of a structure can be improved by raising its fundamental eigenfrequency. Thus, we have a lower bound on its eigenvalues,

$$\lambda_{\min}(E) \ge r.$$

Let n be the number of nodes (vertices of the elements). Let nig denote the number of Gauss integration points in each element. In every element, the displacement vector $u(x)$ is approximated as a continuous function, which is linear in every coordinate,

$$u(x) = \sum_{i=1}^n u_i \vartheta_i(x),$$

where u_i is the value of u at the ith node, and ϑ_i is the basis function associated with the ith node. For ϑ_j, define matrices

$$\hat{B}_j \stackrel{\text{def}}{=} \begin{pmatrix} \frac{\partial \vartheta_j}{\partial x_1} & 0 \\ 0 & \frac{\partial \vartheta_j}{\partial x_2} \\ \frac{1}{2}\frac{\partial \vartheta_j}{\partial x_2} & \frac{1}{2}\frac{\partial \vartheta_j}{\partial x_1} \end{pmatrix} \text{ (for two dimensions),}$$

$$\hat{B}_j \stackrel{\text{def}}{=} \begin{pmatrix} \frac{\partial \vartheta_j}{\partial x_1} & 0 & 0 \\ 0 & \frac{\partial \vartheta_j}{\partial x_2} & 0 \\ 0 & 0 & \frac{\partial \vartheta_j}{\partial x_3} \\ \frac{1}{2}\frac{\partial \vartheta_j}{\partial x_2} & \frac{1}{2}\frac{\partial \vartheta_j}{\partial x_1} & 0 \\ 0 & \frac{1}{2}\frac{\partial \vartheta_j}{\partial x_3} & \frac{1}{2}\frac{\partial \vartheta_j}{\partial x_2} \\ \frac{1}{2}\frac{\partial \vartheta_j}{\partial x_3} & 0 & \frac{1}{2}\frac{\partial \vartheta_j}{\partial x_1} \end{pmatrix} \text{ (for three dimensions).}$$

For Ω_i, let $B_{i,k}$ be the block matrix whose jth block is $\hat{B}_j$ evaluated at the kth integration point and zero otherwise. The full dimension of $B_{i,k}$ is $3 \times 2n$ for the two-dimensional case and $6 \times 3n$ for the three-dimensional case.

Let $A(E)$ denote the *stiffness matrix* relating the forces to the displacements, and let $A_i \in \mathbb{R}^{N\times N}$ denote the *element stiffness matrices*

$$A(E) \stackrel{\text{def}}{=} \sum_{i=1}^{m} A_i(E), \qquad A_i(E) = \sum_{k=1}^{nig} B_{i,k}^\top E_i B_{i,k}.$$

Since the material obeys the Hooke's law, forces (loads) on each element, denoted as f_j $(j = 1, \ldots, L)$, are linearly related to the displacement vector,

$$f_j = A(E)u, \qquad j = 1, \ldots, L. \tag{1}$$

The system is in equilibrium for u if outer and inner forces balance each other. The equilibrium is measured by the *compliances* of the system: the lower the compliance, the more rigid the structure with respect to the loads. The compliance can be represented as

$$f_j^\top u.$$

In the minimum-cost FMO model, an upper bound $\gamma > 0$ is imposed on the compliances. Further, in view of (1), we have

$$\langle A(E)^{-1} f_j, f_j\rangle \le \gamma, \quad j = 1, \ldots, L.$$

In summary, with given loads $f_j, (j = 1, \ldots, L)$, imposed upper and lower bounds $\rho_l^{(i)}$ and $\rho_u^{(i)}$ $(i = 1, \ldots, m)$, r, and compliance upper bound γ, the minimum-cost multiple-load material design problem is the following:

$$\begin{array}{ll} \min\limits_{E\in S_k^{+m}} & \sum\limits_{i=1}^{m} \langle I_k, E_i\rangle \\ \text{s.t.} & \rho_l^{(i)} \le \langle I_k, E_i\rangle \le \rho_u^{(i)}, \ i = 1, \ldots, m, \\ & \langle A(E)^{-1} f_j, f_j\rangle \le \gamma, \ j = 1, \ldots, L, \\ & \lambda_{\min}(E) \ge r. \end{array} \tag{2}$$

Some optimization approaches have been applied to FMO; for instance, Zowe, Kočvara, and Bendsøe [24] formulate the multiple-load FMO as a max-min convex program. They propose penalty/barrier multiplier methods and interior-point methods for the problem. Ben-Tal et al. [2] consider the bounded trace minimum compliance multiple-load FMO problem. They formulate the problem as a semidefinite program and solve it by an interior-point method. Stingl, Kočvara, and Leugering [21] solve the minimum compliance multiple-load FMO problem by a sequential semidefinite programming algorithm. Weldeyesus and Stolpe [22] propose a primal-dual interior-point method to several equivalent FMO formulations. Stingl, Kočvara, and Leugering [20] study the minimum compliance single-load FMO problem with vibration constraint and propose an approach based on nonlinear semidefinite low-rank approximation of the semidefinite dual. Haslinger et al. [8] extend the original problem statement by a class of generic constraints. Czarnecki and Lewiński [5] deal with minimization of the weighted sum of compliances related to the nonsimultaneously applied load cases. All of these approaches are second-order methods. To the best of our knowledge, no first-order methods have been applied to FMO.

Second-order methods exploit the information of Hessians in addition to gradients and function values. Thus, compared with first-order methods, second-order methods generally converge faster and are more accurate; on the other hand, first-order methods do not require formulation, storage, and inverse of Hessian and thus can be applied to large-scale problems. For certain structured problems with bounded simple feasible sets, Nesterov [13] showed that the complexity of fast gradient methods is one magnitude lower than the theoretical lower complexity bound of the gradient-type method for the black-box oracle model. Afterward, there appeared quite a lot of papers on fast gradient-type methods; see, e.g., [12, 15, 14, 16, 17, 6, 4, 18, 23].

However, not every real-world problem is suitable for second-order methods or fast gradient-type methods, for instance, when the structure of the problem is too complex to allow application of the interior-point method or the smoothing technique. The minimum weight FMO model (2) is such a case. For the model, although the matrices $B_{i,l}$ are sparse, $A(E)$ is generally not. The number m is at least in the thousands, and n is smaller than m only by a constant factor. To roughly measure the amount of work per iteration, we use flops, i.e., floating point operations, such as arithmetic operations $(+, -, *, /, \sqrt{\cdot}, \frac{1}{\cdot})$, comparisons, and exchanges. It takes a vector of length $\frac{N(N+1)}{2}$ to store the matrix $A(E)$ or its Cholesky factor in the memory, and about $\left[(k+\frac{1}{2})nig \cdot mN^2 + \frac{1}{2}N^3\right]$ flops to evaluate $\langle A(E)^{-1}f_j, f_j\rangle$. Hence, it is difficult to manage model (2) of reasonable size by second-order methods, since second-order methods work on the Hessian of the problem whose size is at least the square of total variables. Also, the variables of model (2) are m matrices of size $k \times k$. In addition, the constraints of model (2) are not simple, which prevents us from applying usual gradient-project–type methods to it, because it is not easy to project onto its feasible set.

In this paper, we reformulate model (2) into a saddle-point problem and apply the primal-dual subgradient method [15] to the saddle-point problem. The advantage of our formulation is that the inverse or the Cholesky factorization of $A(E)$ does not need to be calculated, and thus we reduce the computational cost of each iteration to just $\mathcal{O}(N^2)$.

The traditional subgradient method for minimizing a nonsmooth convex function F over the Euclidean space employs a prechosen sequence of steps $\{\lambda_k\}_{k=0}^{\infty}$ which satisfies the divergent-series rule,

$$\lambda_k > 0, \qquad \lambda_k \to 0, \qquad \sum_{k=0}^{\infty} \lambda_k = \infty.$$

The iterates are generated as follows:

$$g_k \in \partial F(x_k), \qquad x_{k+1} = x_k - \lambda_k g_k, \qquad k \geq 0.$$

In the traditional subgradient method, new subgradients enter the model with decreasing weights, contradicting the general principle of iterative schemes which states that new information is more important than old. But the vanishing of steps is necessary for the convergence of the iterates $\{x_k\}_{k=0}^{\infty}$.

The primal-dual subgradient method [15] associates the primal minimization sequence with a master process in the dual space; it does not have the drawback of diminishing step sizes in the dual space; the method is proven to be optimal for saddle-point problems, nonsmooth convex minimization, minimax problems, variational inequalities, and stochastic optimization. Let $\mathcal{E}$ be a finite-dimensional real vector space equipped with a norm $\|\cdot\|$. Let $\mathcal{E}^*$ be its dual. Let $Q \in \mathcal{E}$ be a closed convex set. Let $d(x)$ be a prox-function of Q with convexity parameter $\sigma \geq 0$ satisfying: $\forall\, x, y \in Q$, $\forall \alpha \in [0, 1]$,

$$d\left(\alpha x + (1-\alpha)y\right) \leq \alpha d(x) + (1-\alpha)d(y) - \frac{1}{2}\sigma\alpha(1-\alpha)\left\|x - y\right\|^2.$$

Let $\mathcal{G}$ be a function mapping $\mathcal{E}$ to $\mathcal{E}^*$. For instance, for the convex minimization problem, the function $\mathcal{G}$ can be a subgradient of the objective function. The generic scheme of dual averaging (DA-scheme) [15] works as follows:

Initialization: Set $s_0 = 0 \in \mathcal{E}^*$. Choose $\beta_0 > 0$.
Iteration $(k \geq 0)$:
1. Compute $g_k = \mathcal{G}(x_k)$.
2. Choose $\lambda_k > 0$. Set $s_{k+1} = s_k + \lambda_k g_k$.
3. Choose $\beta_{k+1} \geq \beta_k$. Set $x_{k+1} = \arg\min_{x \in Q} \left\{\langle s_{k+1}, x\rangle + \beta_{k+1} d(x)\right\}$.

Let

$$\hat{\beta}_0 = \hat{\beta}_1, \quad \hat{\beta}_{i+1} = \hat{\beta}_i + \frac{1}{\hat{\beta}_i}, \quad i \geq 1.$$

The scheme has two main variants: simple averages where $\lambda_k = 1$ and $\beta_k = \gamma\hat{\beta}_k$ with constant $\gamma > 0$, and weighted averages where $\lambda_k = 1/\|g_k\|_*$ and $\beta_k = \frac{\hat{\beta}_k}{\rho\sqrt{\sigma}}$ with constant $\rho > 0$.

There are other gradient methods for saddle-point problems. In [4], Chambolle and Pock study a first-order primal-dual algorithm for a class of saddle-point problems in two finite-dimensional real vector spaces $\mathcal{E}$ and $\mathcal{V}$,

$$\min_{x \in \mathcal{E}} \max_{y \in \mathcal{V}} \langle Kx, y\rangle + G(x) - T^*(y),$$

where $K\colon \mathcal{E} \to \mathcal{V}$ is a linear operator, and G and T^* are proper convex, lower-semicontinuous functions. That algorithm, as well as the classical Arrow–Hurwicz method [1] and its variants for saddle-point problems, is not applicable to our FMO formulation, because in our formulation the function between two spaces is nonlinear. Nemirovski's prox-method [12] reduces the problem of approximating a saddle-point of a $C^{1,1}$ function to that of solving the associated variational inequality by a prox-method. The approach is not applicable to our FMO formulation, because the structure of our FMO formulation is not simple enough and its objective function is not in $C^{1,1}$.

In our approach, the inverse of $A(E)$ in model (2) does not need to be calculated, which decreases computational cost per iteration by one order of magnitude. Solutions of the primal and dual subproblems at each iteration can be written in closed form. Each iteration takes roughly $(6k \cdot nig \cdot L)mN$ flops. The auxiliary storage space is linear in N. Furthermore, the primal subproblem is decoupled into m small problems that can be solved in parallel. Each small problem can be solved in approximately $(10k^3)$ flops. Thus, it is possible to work on large-scale problems, compared with second-order methods dealing with the Hessian of $6m$ or $21m$ variables plus additional constraints on the m matrices. To prove the efficiency of the algorithm, we give iteration complexity bounds of our algorithm, which includes *simple dual averaging* and *weighted dual averaging* schemes. The complexity bounds are optimal for the general subgradient methods. Numerical experiments are described at the end of the paper.

The remainder of the paper is organized as follows. In section 2, we give our saddle-point form of the problem. In section 3, we show that a solution to our bounded Lagrangian form either solves the original problem or gives an approximate solution of the original problem. In section 4, we present our algorithm. In section 5, we give closed-form solutions to the subproblems. In section 6, we derive complexity bounds of our algorithm. In section 8, we present some computational examples of our algorithm. In section 7, we describe and analyze a penalized Lagrangian approach. In the appendices, we give a closed-form solution of a related matrix projection problem and an update scheme for the parameters of the algorithm.

2. Saddle-point formulation.

We first rewrite problem (2) in a saddle-point form. Define

$$Q_k^{(i)} \stackrel{\text{def}}{=} \left\{ U \in S_k \colon \lambda_{\min}(U) \geq r,\ \rho_l^{(i)} \leq \langle I_k, U \rangle \leq \rho_u^{(i)} \right\}. \tag{3}$$

The second group of constraints in (2) can be represented in max form as

$$\gamma \;\geq\; \langle A(E)^{-1} f_j, f_j \rangle \quad = \quad \max_{y_j \in \mathbb{R}^N} \left\{ 2\langle f_j, y_j \rangle - \langle A(E) y_j, y_j \rangle \right\}.$$

Assume that problem (2) satisfies some constraint qualifications such as the Slater condition—there exists $\hat{E}$ such that $\langle A(\hat{E})^{-1} f_j, f_j \rangle < \gamma$ for $j = 1, \ldots, L$. Then a Lagrangian multiplier exists, and we can solve the Lagrangian of problem (2) instead.

Thus, problem (2) can be written as follows:

$$\min_{\substack{E_i \in Q_k^{(i)}\\ k=1,\dots,m}} \max_{\substack{y_j \in \mathbb{R}^N, \lambda_j \geq 0\\ j=1,\dots,L}} \left\{ \sum_{i=1}^{m} \langle I_k, E_i\rangle + \sum_{j=1}^{L} \lambda_j \cdot [2\langle f_j, y_j\rangle - \langle A(E) y_j, y_j\rangle - \gamma] \right\}$$

$$\overset{\lambda_j y_j \to x_j}{=} \min_{\substack{E_i \in Q_k^{(i)}\\ k=1,\dots,m}} \left\{ \sum_{i=1}^{m} \langle I_k, E_i\rangle \right.$$

$$\left. + \max\left(0, \max_{\substack{x_j \in \mathbb{R}^N, \lambda_j > 0\\ j=1,\dots,L}} \sum_{j=1}^{L} \left[2\langle f_j, x_j\rangle - \frac{1}{\lambda_j}\langle A(E) x_j, x_j\rangle - \gamma\lambda_j \right] \right) \right\}$$

$$= \min_{\substack{E_i \in Q_k^{(i)}\\ k=1,\dots,m}} \max_{\substack{x_j \in \mathbb{R}^N,\\ j=1,\dots,L}} \left\{ \sum_{i=1}^{m} \langle I_k, E_i\rangle + \sum_{j=1}^{L} 2\left[\langle f_j, x_j\rangle - \gamma^{1/2}\langle A(E) x_j, x_j\rangle^{1/2} \right] \right\}.$$

The dimension of the matrix $A(E)$ is large; the first transformation eliminates the need for calculating its inverse, but that results in a nonconcave objective function in λ and y. The second transformation makes the function concave in λ and x. In the last step, variable λ is eliminated to simplify the formulation.

Define

$$F(E, x) \stackrel{\text{def}}{=} \sum_{i=1}^{m} \langle I_k, E_i\rangle + \sum_{j=1}^{L} 2\left[\langle f_j, x_j\rangle - \gamma^{1/2}\langle A(E) x_j, x_j\rangle^{1/2} \right].$$

Thus, to solve problem (2), we only need to solve

$$\min_{\substack{E_i \in Q_k^{(i)}\\ i=1,\dots,m}} \max_{\substack{x_j \in \mathbb{R}^N,\\ j=1,\dots,L}} F(E, x). \tag{4}$$

Note that $F(E, x)$ is convex in E and concave in $x \in \mathbb{R}^{L \times N}$.

3. Bounded Lagrangian. We apply the primal-dual subgradient method [15] to the saddle-point formulation (4). The convergence of the algorithm requires the iterates to be uniformly bounded [15]. We therefore impose a bound on x:

$$\min_{\substack{E_i \in Q_k^{(i)}\\ i=1,\dots,m}} \max_{\substack{\|x_j\| \leq \eta,\\ j=1,\dots,L}} F(E, x). \tag{5}$$

Next we show that the primal solution of the saddle-point problem (5) is either a solution to the original problem (2) or an approximate solution in the sense that its constraint-violation is bounded by η^{-1} and its objective value is smaller than that of the optimal value of (2).

Let (E^*, x^*) be a solution to the saddle-point problem (4); then for any $\alpha \geq 0$, $(E^*, \alpha x^*)$ is also its solution. We can choose α^* small enough; for instance, let $\alpha^* = \eta / \max\{\|x_j^*\|, 1\}$, so that $(E^*, \alpha^* x^*)$ is a solution to the bounded saddle-point form (5).

For any $E \in Q_k$, denote the index set of its violated constraints as

$$W_E \stackrel{\text{def}}{=} \left\{1 \le j \le L \colon \langle f_j, A(E)^{-1} f_j\rangle > \gamma\right\}.$$

Define $F(E) \stackrel{\text{def}}{=} \max_x F(E,x)$. We have the following results regarding our material design problem.

LEMMA 1. *Let $(\tilde{E},\tilde{x})$ be a solution of (5). Let F^* be the optimal value of (2).*

1. *If $\|\tilde{x}_j\| < \eta$ for $j = 1,\dots,L$, then $\tilde{E}$ is a solution to (2).*
2. *Otherwise, $\tilde{E}$ has the following properties:*
   (a) $F(\tilde{E}) \le F^*$.
   (b) $\sum_{j\in W_{\tilde{E}}} (\langle f_j, A(\tilde{E})^{-1} f_j\rangle^{1/2} - \gamma^{1/2}) \le \frac{F^* - m\rho_l}{2r\lambda_{\min}(BB^T)\eta}$.

Proof. Item 1 is obvious, as the constraints are nonbinding.

Next, we prove item 2.

Because

$$\max_{\substack{\|x_j\|\le\eta\\ j=1,\dots,L}} F(E,x) \le \max_{\substack{x_j\in\mathbb{R}^N,\\ j=1,\dots,L}} F(E,x),$$

we have item 2(a). For any fixed $E \in Q$, the point

$$x_j = \begin{cases} \frac{\eta}{\|A(E)^{-1}f_j\|} A(E)^{-1} f_j, & j \in W_E,\\ 0, & j \notin W_E \end{cases}$$

is feasible for

$$\max_{\substack{\|x_j\|\le\eta,\\ j=1,\dots,L}} F(E,x)$$

with objective value

$$F_x(E) \stackrel{\text{def}}{=} \langle I, E\rangle + 2\sum_{j\in W_E} \left(\langle f_j, A(E)^{-1} f_j\rangle^{1/2} - \gamma^{1/2}\right) \frac{\langle f_j, A(E)^{-1} f_j\rangle^{1/2}}{\|A(E)^{-1}f_j\|}\eta. \tag{6}$$

Since

$$\langle f_j, A(E)^{-1} f_j\rangle = \langle A(E)^{-1} f_j, A(E)(A(E)^{-1} f_j)\rangle,$$

we also have

$$\frac{\langle f_j, A(E)^{-1} f_j\rangle^{1/2}}{\|A(E)^{-1}f_j\|} \ge \lambda_{\min} A(E)^{1/2} \ge r\lambda_{\min}(BB^T)$$

and

$$\langle I, E\rangle \ge m\rho_l.$$

Therefore,

$$F(\tilde{E},\tilde{x}) \ge m\rho_l + 2r\lambda_{\min}(BB^T)\eta \textstyle\sum_{j\in W_{\tilde{E}}} \left(\langle f_j, A(\tilde{E})^{-1} f_j\rangle^{1/2} - \gamma^{1/2}\right).$$

By item 2(a) of the lemma, we have

$$F(\tilde{E},\tilde{x}) \le F^*.$$

Item 2(b) then follows. ∎

Note that as $\eta \to +\infty$, the set of saddle-points of (5) approaches that of the original problem.

4. The algorithm. In this section, we describe how to apply the primal-dual subgradient method [15] to the saddle-point reformulation of model (2). We have developed a parameter update scheme for the algorithm, which is included in Appendix B.

For a matrix V, let vector $\lambda(V)$ denote the eigenvalues of V; let $\lambda_{\min}(V)$ be the smallest eigenvalue of V. The gradient (subgradients) of $F(E,x)$ at (E,x) are as follows: for $i = 1,\dots,m$, $j = 1.\dots,L$,

$$g_{E_i}(E,x) = I_k - \sqrt{\gamma}\sum_{j\in R}\langle A(E)x_j, x_j\rangle^{-1/2}\left(\sum_{l=1}^{nig} B_{i,l}x_jx_j^TB_{i,l}^T\right),$$

$$\text{where } R \stackrel{\text{def}}{=} \{1 \le l \le L\colon \langle A(E)x_l, x_l\rangle > 0\};$$

$$g_{x_j}(E,x) = \begin{cases} 2f_j - 2\sqrt{\gamma}\langle A(E)x_j, x_j\rangle^{-1/2}A(E)x_j, & \langle A(E)x_j, x_j\rangle > 0, \\ \{2f_j - 2\sqrt{\gamma}A(E)y\colon \langle A(E)y, y\rangle = 1\}, & \langle A(E)x_j, x_j\rangle = 0. \end{cases}$$

For the primal space, we choose the standard Frobenius norm,

$$\|E\|_F^2 = \sum_{i=1}^m \|E_i\|_F^2 = \operatorname{tr}(E^2), \qquad d(E) = \frac{1}{2}\sum_{i=1}^m \|E_i - rI_k\|_F^2.$$

For the dual space, we choose the standard Euclidean norm,

$$\|x\|_2^2 = \sum_{j=1}^L \|x_j\|_2^2 = x^Tx, \qquad d(x) = \frac{1}{2}\sum_{j=1}^L \|x_j\|_2^2.$$

Their dual norms are denoted as $\|\cdot\|_{F,*} = \|\cdot\|_F$, $\|\cdot\|_{2,*} = \|\cdot\|_2$.

The set $Q_k^{(i)}$ for E_i is defined in (3), and the set Q_x for x_j is

$$Q_x = \left\{x_j \in \mathbb{R}^N\colon \|x_j\| \le \eta\right\}.$$

Note that F is nonsmooth. The primal-dual subgradient method [15] for saddle-point problems (5) works as follows.

Initialization: Set $s_0^{E_i} = 0\,(i = 1,\dots,m)$, $s_0^{x_j} = 0\,(j = 1,\dots,L)$.
Choose $\beta_0 > 0$, $0 < \tau < 1$.

Iteration $t = 0, 1, \dots$
1. Compute $g_{E_i}^{(t)}(E^{(t)}, x^{(t)})$, $g_{x_j}^{(t)}(E^{(t)}, x^{(t)})$, for $i = 1,\dots,m; j = 1,\dots,L$.
2. Choose $\alpha_t > 0$, set
$s_{t+1}^{E_i} = s_t^{E_i} + \alpha_t g_{E_i}^{(t)}\,(i = 1,\dots,m), \quad s_{t+1}^{x_j} = s_t^{x_j} - \alpha_t g_{x_j}^{(t)}\,(j = 1,\dots,L)$.
3. Choose $\beta_{t+1} \ge \beta_t$, set
$E^{(t+1)} = \arg\min_{E_i\in Q_k^{(i)}} \left\{\langle s_{t+1}^E, E\rangle + \beta_{t+1}\tau d_E(E)\right\}$,
$x^{(t+1)} = \arg\min_{x_j\in Q_x} \left\{\langle s_{t+1}^x, x\rangle + \beta_{t+1}(1-\tau)d_x(x)\right\}$.

Output: $\hat{E}^{(t+1)} = \frac{1}{\sum_{l=0}^t \alpha_l}\sum_{l=0}^t \alpha_l E^{(l)}$.

Details of a parameter update scheme for β_t are given in Appendix B.

We take

$$\hat{\beta}_0 = \hat{\beta}_1 = 1, \quad \hat{\beta}_{t+1} = \hat{\beta}_t + \frac{1}{\hat{\beta}_t}, \ t = 1, \ldots,$$

$$\beta_t = \sigma \hat{\beta}_t, \ t = 0, \ldots.$$

Based on different choices of α, there are two variants of the algorithm as follows:

1. *Method of simple dual averages.*
   We let
   $$\alpha_t = 1, \ t = 0, \ldots.$$
2. *Method of weighted dual averages.*
   We let
   $$\alpha_t = 1 \Big/ \left(\frac{\|g_E^{(t)}\|_{F,*}^2}{\tau} + \frac{\|g_x^{(t)}\|_{2,*}^2}{1-\tau} \right)^{1/2}, \quad t = 0, \ldots.$$

5. Solution to the subproblem. In this section, we give closed-form solutions to the subproblems at each iteration of our algorithm.

Solution of x. The closed-form solution for $x^{(t+1)}$ in step 3 of the algorithm is derived as below.

By the Cauchy–Schwarz–Bunyakovsky inequality, for $j = 1, \ldots, L$,

$$\begin{aligned} & \langle s_{t+1}^{x_j}, x_j \rangle + \beta_{t+1}(1-\tau) d_{x_j}(x_j) \\ \geq \ & -\|s_{t+1}^{x_j}\|_{2,*} \cdot \|x_j\|_2 + \frac{\beta_{t+1}(1-\tau)}{2} \|x_j\|_2^2 \\ = \ & \frac{\beta_{t+1}(1-\tau)}{2} \left(\|x_j\|_2 - \frac{1}{\beta_{t+1}(1-\tau)} \|s_{t+1}^{x_j}\|_{2,*} \right)^2 - \frac{1}{2\beta_{t+1}(1-\tau)} \|s_{t+1}^{x_j}\|_{2,*}^2, \end{aligned}$$

with equality iff $x_j = -\nu s_{t+1}^{x_j}$ for some $\nu \geq 0$. Therefore,

$$x_j^{(t+1)} = -\min \left(\frac{\eta}{\|s_{t+1}^{x_j}\|_{2,*}}, \frac{1}{\beta_{t+1}(1-\tau)} \right) s_{t+1}^{x_j}. \tag{7}$$

Solution of E. For a set M, let $|M|$ denote the cardinality of M, i.e., the number of elements in M. In step 3 of the algorithm, $E_i^{(t+1)}$ can be seen as the projection

$$\min_{V \in Q_k^{(i)}} \left\| V + \frac{1}{\beta_{t+1}\tau} s_{t+1}^{E_i} - rI \right\|_F^2.$$

By Theorem 15 in Appendix A, we can represent E^{t+1} as follows.

For each $1 \leq i \leq m$, let $U \Lambda U^T$ be the eigenvalue decomposition of $s_{t+1}^{E_i}$, and let $\lambda_1, \ldots, \lambda_k$ be its eigenvalues. Define the sets

$$M_0 \stackrel{\text{def}}{=} \{1 \leq l \leq k \colon \lambda_l \geq 0\}, \quad \bar{M}_0 \stackrel{\text{def}}{=} \{1, \ldots, k\} \setminus M_0.$$

Then

$$E_i^{(t+1)} = U \operatorname{diag}(\omega) U^T, \tag{8}$$

where ω is determined according to the following three cases.

1. $\beta_{t+1}\tau(kr-\rho_u^{(i)}) \le \sum_{q\in\bar{M}_0}\lambda_q \le \beta_{t+1}\tau(kr-\rho_l^{(i)})$.
   Let
$$\omega_l \quad = \quad \begin{cases} r, & l\in M_0,\\ r-\frac{\lambda_l}{\beta_{t+1}\tau}, & l\notin M_0.\end{cases}$$

2. $\sum_{q\in\bar{M}_0}\lambda_q < \beta_{t+1}\tau(kr-\rho_u^{(i)})$.
   Then there is a partition $\bar{M}_0 = P\cup\bar{P}$,
$$P \quad = \quad \left\{ l\in\bar{M}_0\colon \lambda_l < \frac{\beta_{t+1}\tau(\rho_u^{(i)}-kr)+\sum_{q\in P}\lambda_q}{|P|}\right\},$$
$$\bar{P} \quad = \quad \left\{ l\in\bar{M}_0\colon \lambda_l \ge \frac{\beta_{t+1}\tau(\rho_u^{(i)}-kr)+\sum_{q\in P}\lambda_q}{|P|}\right\}.$$
   Let
$$\omega_l = \begin{cases} r, & l\in\bar{P}\cup M_0,\\ r-\frac{\lambda_l}{\beta_{t+1}\tau}+\frac{\beta_{t+1}\tau(\rho_u^{(i)}-kr)+\sum_{q\in P}\lambda_q}{\beta_{t+1}\tau|P|}, & l\in P.\end{cases}$$

3. $\sum_{q\in\bar{M}_0}\lambda_q > \beta_{t+1}\tau(kr-\rho_l^{(i)})$.
   Then there is a partition $M_0 = P_m\cup\bar{P}_m$,
$$P_m \quad = \quad \left\{ l\in M_0\colon \frac{\rho_l^{(i)}+\frac{1}{\beta_{t+1}\tau}\sum_{j\in P_m\cup\bar{M}_0}\lambda_j - kr}{|P_m|+|\bar{M}_0|} > \frac{\lambda_l}{\beta_{t+1}\tau}\right\}.$$
   Let
$$\omega_l \quad = \quad \begin{cases} -\frac{\lambda_l}{\beta_{t+1}\tau}+\frac{\beta_{t+1}\tau(\rho_l^{(i)}-|\bar{P}_m|r)+\sum_{q\in\bar{M}_0\cup P_m}\lambda_q}{\beta_{t+1}\tau(|\bar{M}_0|+|P_m|)}, & l\in\bar{M}_0\cup P_m,\\ r, & l\in\bar{P}_m.\end{cases}$$

The eigenvalues ω in case 2 can be obtained by the following algorithm.

ALGORITHM PROJSYML.

Step 1 (Initialization) Let $\lambda_{\sigma(1)}\le\cdots\le\lambda_{\sigma(p)}<0$ be the p negative eigenvalues of $s_{t+1}^{E_i}$.
Let
$$P=\{\sigma(1)\},\quad T=\beta_{t+1}\tau(\rho_u^{(i)}-kr)+\lambda_{\sigma(1)},\quad q=1.$$

Step 2. While $q\lambda_{\sigma(q+1)}<T$, do
$$P\cup\{\sigma(q+1)\}\to P,\quad T+\lambda_{\sigma(q+1)}\to T,\quad q+1\to q.$$

Step 3. Let
$$\omega_l \quad = \quad \begin{cases} r, & l\notin P,\\ r-\frac{\lambda_l}{\beta_{t+1}\tau}+\frac{T}{\beta_{t+1}\tau q}, & l\in P.\end{cases}$$

Similarly, the eigenvalues in case 3 can be obtained by the following algorithm.

ALGORITHM PROJSYMG.

Step 1 (Initialization) Let $0<\lambda_{\sigma(1)}\le\cdots\le\lambda_{\sigma(u)}$ be the u positive eigenvalues of $s_{t+1}^{E_i}$.

- If $u = p$, let

$$U = \{\sigma(1)\}, \quad T = \beta_{t+1}\tau(\rho_l^{(i)} - kr) + \lambda_{\sigma(1)}, \quad q = 1.$$

- If $u < p$, let

$$U = \bar{M}_0 \cup \{i\colon \lambda_i = 0\}, \quad T = \beta_{t+1}\tau(\rho_l^{(i)} - kr) + \sum_{j \in \bar{M}_0} \lambda_j, \quad q = |U|.$$

Step 2 While $q\lambda_{\sigma(q+1)} < T$, do

$$U \cup \{\sigma(q+1)\} \to U, \quad T + \lambda_{\sigma(q+1)} \to T, \quad q + 1 \to q.$$

Step 3 Let

$$\omega_l = \begin{cases} r, & l \in M_0 \setminus U, \\ r - \frac{\lambda_l}{\beta_{t+1}\tau} + \frac{T}{\beta_{t+1}\tau q}, & l \in \bar{M}_0 \cup U. \end{cases}$$

6. Complexity of the algorithm. To understand the complexity of the algorithm for model (2), in this part we study duality gap and computational cost of each iteration. By [15], it takes $\mathcal{O}(\frac{1}{\epsilon^2})$ iterations to solve a general convex-concave saddle-point problem to the absolute accuracy ϵ, which is the exact lower complexity bound for such a class of algorithm schemes. To give insight into how the data of an FMO model, such as f, B, and η, affect convergence time, we give upper bounds of the duality gap of the iterates generated by our algorithm in terms of the number of iterations and input data in section 6.1. In section 6.2, we derive computational cost per iteration. From the duality gap and computational cost per iteration given in this section, we can estimate from given data an upper bound on computational effort needed to approximate a solution of a problem instance of model (2) based on the method proposed in the paper.

6.1. Iteration bounds. By [7, Chapter 6, Proposition 2.1], for a function $L\colon \mathcal{A} \times \mathcal{B} \mapsto \mathbb{R}$, assume that

- the sets $\mathcal{A}$ and $\mathcal{B}$ are convex, closed, nonempty, and bounded;
- for any fixed $u \in \mathcal{A}$, $p \mapsto L(u,p)$ is concave and upper semicontinuous;
- for any fixed $p \in \mathcal{A}$, $u \mapsto L(u,p)$ is convex and upper semicontinuous;

then the function L has at least one saddle-point.

Since E and x are bounded, and F is continuous and finite, by the above results, we conclude that F has a saddle-point and a finite saddle-value. An upper bound on duality gap is given in [15, Theorem 6]. We next represent the duality gap in terms of input data.

Define

$$\|(g_E, g_x)\|_* \stackrel{\text{def}}{=} \left[\frac{1}{\tau}\|g_E\|_{F,*}^2 + \frac{1}{1-\tau}\|g_x\|_{2,*}^2\right]^{1/2},$$

$$\|(E, x)\| \stackrel{\text{def}}{=} \left[\tau\|E\|_F^2 + (1-\tau)\|x\|_2^2\right]^{1/2}.$$

For a matrix V, denote $\|V\|_2^2 \stackrel{\text{def}}{=} \lambda_{\max}(V^T V)$.

Since

$$\begin{pmatrix} \rho_u - (k-1)r & & & \\ & r & & \\ & & \ddots & \\ & & & r \end{pmatrix} \in \arg\max_{E_i \in Q_k^{(i)}} d_E(E_i),$$

we have

$$D_E \overset{\text{def}}{=} \frac{1}{2}\max_{E_i\in Q_k^{(i)}} \|E-rI\|_F^2 \le \frac{1}{2}m(\rho_u-kr)^2. \tag{9}$$

Furthermore, by our algorithm scheme,

$$D_x \overset{\text{def}}{=} \max_{x\in Q_x} \tfrac{1}{2}\|x\|_2^2 \le \tfrac{L}{2}\eta^2. \tag{10}$$

Define

$$\kappa_t \overset{\text{def}}{=} \frac{1}{\sum_{l=0}^t \alpha_l} \max_{E_i\in Q_k^{(i)}} \left\{\sum_{l=0}^t \alpha_l \left\langle g_E\left(E^{(l)},x^{(l)}\right), E^{(l)}-E\right\rangle\right\},$$

$$\upsilon_t \overset{\text{def}}{=} \frac{1}{\sum_{l=0}^t \alpha_l} \max_{x_j\in\mathbb{R}^N} \left\{\sum_{l=0}^t \alpha_l \left\langle g_x\left(E^{(l)},x^{(l)}\right), x-x^{(l)}\right\rangle : \|x_j\|_2\le\eta\right\}.$$

By the Cauchy–Schwarz–Bunyakovsky inequality, it is easy to verify that

$$\upsilon_t = \frac{1}{\sum_{l=0}^t \alpha_l}\left(\sum_{j=1}^L \eta\|s_{t+1}^{x_j}\|_{2,*} - \sum_{l=0}^t \alpha_l \left\langle g_x\left(E^{(l)},x^{(l)}\right), x^{(l)}\right\rangle\right),$$

which is attained at

$$x_j = \begin{cases} \frac{\eta}{\|s_{t+1}^{x_j}\|_{2,*}} s_{t+1}^{x_j}, & s_{t+1}^{x_j}\neq 0,\\ 0 & \text{otherwise.}\end{cases}$$

Now let us give a bound for κ_t. Let $\kappa_t = \frac{1}{\sum_{l=1}^t \alpha_l}\sum_{i=1}^m \kappa_i^t$, where

$$\kappa_i^t \overset{\text{def}}{=} \max_{E_i\in Q_k^{(i)}} \left\{\sum_{l=0}^t \alpha_l \left\langle g_{E_i}\left(E^{(l)},x^{(l)}\right), E_i^{(l)}-E_i\right\rangle\right\}.$$

By the Hoffman–Wielandt theorem,

$$\begin{aligned}\kappa_i^t &= \sum_{l=0}^t \alpha_l \langle g_{E_i}(E^{(l)},x^{(l)}), E_i^l\rangle - \min_{E_i\in Q_k^{(i)}} \langle s_{t+1}^{E_i}, E_i\rangle\\ &= \sum_{l=0}^t \alpha_l \langle g_{E_i}(E^{(l)},x^{(l)}), E_i^l\rangle\\ &\quad - \begin{cases}[\rho_l-kr]_+\lambda_{\min}(s_{t+1}^{E_i}) + r\,\mathrm{tr}(s_{t+1}^{E_i}), & \lambda_{\min}(s_{t+1}^{E_i})>0,\\ (\rho_u-kr)\lambda_{\min}(s_{t+1}^{E_i}) + r\,\mathrm{tr}(s_{t+1}^{E_i}), & \lambda_{\min}(s_{t+1}^{E_i})\le 0.\end{cases}\end{aligned}$$

Define

$$\delta_t \overset{\text{def}}{=} \max_{E_i\in Q_k^{(i)}, x\in Q_x} \left\{\sum_{l=0}^t \alpha_l \langle (g_E^{(l)}, g_x^{(l)}), (E^{(l)},x^{(l)})-(E,x)\rangle : d(x)\le\tau D_E + (1-\tau)D_x\right\}.$$

By [15, Theorem 6], $\kappa_t + \upsilon_t$ is a bound of the duality gap; i.e.,

$$0 \leq \max_{x_j \in Q_x} F(\hat{E}^{(t+1)}, x) - \min_{E_i^{(i)} \in Q_k^{(i)}} F(E, \hat{x}^{(t+1)}) \quad \leq \quad \kappa_t + \upsilon_t \leq \frac{1}{\sum_{l=0}^{t} \alpha_l} \delta_t. \tag{11}$$

Next, we bound the above duality gap by input data. To this end, we first bound the partial derivatives g_E and g_x.

Define

$$B_i \stackrel{\text{def}}{=} \begin{bmatrix} B_{i,1} \\ \vdots \\ B_{i,nig} \end{bmatrix}, \qquad B \stackrel{\text{def}}{=} \begin{bmatrix} B_1 \\ \vdots \\ B_m \end{bmatrix}. \tag{12}$$

LEMMA 2. *The partial derivative of $F(E, x)$ in E can be bounded by $\|x\|_2$ as follows:*

$$\|g_E(E, x)\|_{F,*}^2 \quad \leq \quad mk + L^2 \frac{\gamma}{r} \|B\|_2^2 \eta^2 \stackrel{\text{def}}{=} L_E^2.$$

Proof. We have

$$\begin{aligned} \langle A(E)x_j, x_j \rangle &= \sum_{i=1}^{m} \sum_{l=1}^{nig} \langle E_i B_{i,l} x_j, B_{i,l} x_j \rangle \\ &\geq \sum_{i=1}^{m} \lambda_{min}(E_i) \sum_{l=1}^{nig} \langle B_{i,l} x_j, B_{i,l} x_j \rangle \\ &\geq r \sum_{i=1}^{m} \sum_{l=1}^{nig} \langle B_{i,l} x_j, B_{i,l} x_j \rangle, \end{aligned} \tag{13}$$

where the last inequality is from the definition of the set $Q_k^{(i)}$.

Since for two matrices A and B of proper dimensions, $\text{tr}(AB) = \text{tr}(BA)$, we have

$$\begin{aligned} \left\| \sum_{l=1}^{nig} B_{i,l} x_j x_j^T B_{i,l}^T \right\|_F &\leq \sum_{l=1}^{nig} \left\| B_{i,l} x_j x_j^T B_{i,l}^T \right\|_F \\ &= \sum_{l=1}^{nig} \left[\text{tr} \left(B_{i,l} x_j x_j^T B_{i,l}^T B_{i,l} x_j x_j^T B_{i,l}^T \right) \right]^{1/2} \\ &= \sum_{l=1}^{nig} \left[\left(x_j^T B_{i,l}^T B_{i,l} x_j \right)^2 \right]^{1/2} \\ &= \sum_{l=1}^{nig} \langle B_{i,l} x_j, B_{i,l} x_j \rangle. \end{aligned} \tag{14}$$

Therefore,

$$\left\| \langle A(E)x_j, x_j\rangle^{-1/2} \left(\sum_{l=1}^{nig} B_{i,l}x_j x_j^T B_{i,l}^T\right)\right\|_F \leq \frac{1}{\sqrt{r}}\left[\sum_{l=1}^{nig}\langle B_{i,l}x_j, B_{i,l}x_j\rangle\right]^{1/2}$$

$$\leq \frac{1}{\sqrt{r}}\lambda_{\max}\left(\sum_{l=1}^{nig} B_{i,l}^T B_{i,l}\right)^{1/2} \|x_j\|_2.$$

Note that

$$\lim_{x_j\to 0}\left\| \langle A(E)x_j, x_j\rangle^{-1/2} \left(\sum_{l=1}^{nig} B_{i,l}x_j x_j^T B_{i,l}^T\right)\right\|_F = 0.$$

We also have

$$\sum_{i=1}^{m}\sum_{l=1}^{nig} B_{i,l}^T B_{i,l} = B^T B, \qquad \lambda_{\max}\left(\sum_{i=1}^{m}\sum_{l=1}^{nig} B_{i,l}^T B_{i,l}\right)^{1/2} = \|B\|_2.$$

Hence, $g_E(E,x)$ is bounded as below:

$$\begin{aligned}
\|g_E(E,x)\|_{F,*}^2 &\leq \|I\|_F^2 + \gamma \sum_{i=1}^{m}\left\|\sum_{j\in R}\langle A(E)x_j, x_j\rangle^{-1/2}\left(\sum_{i=1}^{nig} B_{i,l}x_j x_j^T B_{i,l}^T\right)\right\|_F^2 \\
&\leq \|I\|_F^2 + L\gamma \sum_{i=1}^{m}\sum_{j\in R}\left\|\langle A(E)x_j, x_j\rangle^{-1/2}\left(\sum_{i=1}^{nig} B_{i,l}x_j x_j^T B_{i,l}^T\right)\right\|_F^2 \\
&= mk + L\gamma \sum_{j\in R}\langle A(E)x_j, x_j\rangle^{-1}\sum_{i=1}^{m}\left\|\sum_{i=1}^{nig} B_{i,l}x_j x_j^T B_{i,l}^T\right\|_F^2 \\
&\overset{(14)}{\leq} mk + L\gamma \sum_{j\in R}\langle A(E)x_j, x_j\rangle^{-1}\sum_{i=1}^{m}\left(\sum_{l=1}^{nig}\langle B_{i,l}x_j, B_{i,l}x_j\rangle\right)^2 \\
&\leq mk + L\gamma \sum_{j\in R}\langle A(E)x_j, x_j\rangle^{-1}\left(\sum_{i=1}^{m}\sum_{l=1}^{nig}\langle B_{i,l}x_j, B_{i,l}x_j\rangle\right)^2 \\
&\overset{(13)}{\leq} mk + L\frac{\gamma}{r}\sum_{j\in R}\sum_{i=1}^{m}\sum_{l=1}^{nig}\langle B_{i,l}x_j, B_{i,l}x_j\rangle \\
&\overset{(12)}{\leq} mk + L^2\frac{\gamma}{r}\|B\|_2^2\eta^2.
\end{aligned} \tag{15}$$

□

Next, we give a bound on the norm of $g_x(E,x)$.

Let $\tilde{E}_i$ be the block diagonal matrix of nig same diagonal blocks E_i. Let $\tilde{E}$ be the block diagonal matrix with diagonal blocks $\tilde{E}_i$, $(i = 1,\ldots,m)$:

$$\tilde{E}_i \overset{\text{def}}{=} \begin{bmatrix} E_i & & \\ & \ddots & \\ & & E_i \end{bmatrix}, \qquad \tilde{E} \overset{\text{def}}{=} \begin{bmatrix} \tilde{E}_1 & & \\ & \ddots & \\ & & \tilde{E}_m \end{bmatrix}.$$

Then

$$A(E) \quad = \quad B^T \tilde{E} B.$$

LEMMA 3. *The partial derivative of $F(E,x)$ in x can be bounded as follows:*

$$\|g_x(E,x)\|_{2,*} \quad \leq \quad 2\|f\|_2 + 2\sqrt{\gamma L(\rho_u - kr + r)}\|B\|_2 \;\stackrel{\text{def}}{=}\; L_x.$$

Proof. For a vector z of proper dimension, we have

$$\|A(E)z\|_2 \quad \leq \quad \|A(E)\|_2^{1/2}\langle A(E)z, z\rangle^{1/2}.$$

For two matrices A and B of proper dimension, it holds that $\lambda_{\max}(AB) \leq \lambda_{\max}(A)\lambda_{\max}(B)$ and $\lambda_{\max}(AB) = \lambda_{\max}(BA)$.

In addition, by the definition of $Q_k^{(i)}$, we have

$$\lambda_{\max}(\tilde{E}) \;=\; \lambda_{\max}(E) \quad \leq \quad \rho_u - (k-1)r.$$

Therefore, $\|A(E)\|_2$ can be bounded as below:

$$\|A(E)\|_2^{1/2} \quad \leq \quad \|\tilde{E}\|_2^{1/2}\|B^T B\|_2^{1/2} \;\leq\; \sqrt{\rho_u - (k-1)r}\|B\|_2.$$

Hence

$$\begin{aligned}\|g_x(E,x)\|_{2,*} \quad &\leq \quad 2\|f\|_2 + 2\sqrt{\gamma}\Bigg(\sum_{j\in R}\langle A(E)x_j, x_j\rangle^{-1}\langle A(E)x_j, A(E)x_j\rangle \\ &+ \quad \sum_{j\notin R}\langle A(E)y, A(E)y\rangle\Bigg)^{1/2} \\ &\leq \quad 2\|f\|_2 + 2\sqrt{\gamma L(\rho_u - kr + r)}\|B\|_2. \end{aligned} \tag{16}$$

∎

Next, we give bounds on the duality gaps.

By [15, Lemma 3], we have

$$\sqrt{2t-1} \quad \leq \quad \hat{\beta}_t \;\leq\; \frac{1}{1+\sqrt{3}} + \sqrt{2t-1}, \quad t \geq 1. \tag{17}$$

THEOREM 4. *If the iterates are generated by the method of* simple dual average*, the duality gap is bounded as*

$$\begin{aligned} 0 \quad &\leq \quad \max_{x_j \in Q_x} F(\hat{E}^{(t+1)}, x) - \min_{E_i \in Q_k^{(i)}} F(E, \hat{x}^{(t+1)}) \\ &\leq \quad \frac{0.37 + \sqrt{2t+1}}{t+1}\Bigg[\sqrt{\left(mk + \frac{\gamma}{r}L^2\|B\|_2^2\eta^2\right)m(\rho_u - kr)} \\ &+ \quad 2\left(\|f\|_2 + \sqrt{\gamma L(\rho_u - kr + r)}\|B\|_2\right)\sqrt{L}\eta\Bigg]. \end{aligned} \tag{18}$$

Proof. Since partial subdifferentials of f are uniformly bounded,

$$\|g_E\|_{F,*} \;\leq\; L_E, \quad \|g_x\|_{2,*} \;\leq\; L_x \quad \forall\, (E,x) \in Q_k \times Q_x,$$

when we choose

$$\frac{1}{\tau} = 1 + \frac{L_x}{L_E}\sqrt{\frac{D_E}{D_x}},$$

$$\sigma = \sqrt{\frac{\tau L_E^2 + (1-\tau)L_x^2}{2\tau D_E + 2(1-\tau)D_x}},$$

by [15, equation (4.6)], we have

$$\frac{1}{\sum_{l=0}^{t}\alpha_l}\delta_t \leq \frac{\hat{\beta}_{t+1}}{t+1}\sqrt{2}\left(L_E\sqrt{D_E} + L_x\sqrt{D_x}\right). \tag{19}$$

Therefore, by (9), (10), (11), (15), (16), (17), we get

$$\begin{aligned} 0 &\leq \max_{x_j\in Q_x} F(\hat{E}^{(t+1)}, x) - \min_{E_i\in Q_k^{(i)}} F(E, \hat{x}^{(t+1)}) \\ &\leq \frac{0.37+\sqrt{2t+1}}{t+1}\left[\sqrt{\left(mk + \frac{\gamma}{r}L^2\|B\|_2^2\eta^2\right)m}(\rho_u - kr)\right. \\ &+ \left. 2\left(\|f\|_2 + \sqrt{\gamma L(\rho_u - kr + r)}\|B\|_2\right)\sqrt{L}\eta\right]. \end{aligned}$$

□

THEOREM 5. *If the iterates are generated by the method of* weighted dual average*, the duality gap is bounded by*

$$\begin{aligned} 0 &\leq \max_{x_j\in Q_x} F(\hat{E}^{(t+1)}, x) - \min_{E_i\in Q_k^{(i)}} F(E, \hat{x}^{(t+1)}) \\ &\leq \min\left\{\frac{0.37+\sqrt{2t+1}}{t+1}\left[\sqrt{m^2k + L^2m\frac{\gamma}{r}\|B\|_2^2\eta^2}(\rho_u - kr) + 2\sqrt{L}\eta\|f\|_2\right.\right. \\ &+ \left. 2\sqrt{\gamma(\rho_u - kr + r)}\|B\|_2 L\eta\right], \ \frac{(4\sqrt{2}+2)\hat{\beta}_{t+1}\sqrt{d(E^*, x^*)}}{t+1} \\ &\times \left[mk + 8(3+\sqrt{2})\tfrac{\gamma}{r}L\|B\|_2^2 d(E^*, x^*)\right. \\ &+ \left.\left. 4\left(\|f\|_2 + \sqrt{\gamma L(\rho_u - kr + r)}\|B\|_2\right)^2\right]^{1/2}\right\}. \end{aligned} \tag{20}$$

Proof.

1. *Bound* 1. Let (E^*, x^*) be an optimal solution. Since

$$d(E, x) = \frac{\tau}{2}\|E - rI\|_F^2 + \frac{1-\tau}{2}\|x\|_2^2, \tag{21}$$

we get

$$\sqrt{d(E,x)} \leq \sqrt{d(E^*, x^*)} + \frac{1}{\sqrt{2}}\|(E,x) - (E^*, x^*)\|.$$

In addition, [15, Theorem 3] states that

$$\|(E,x) - (E^*, x^*)\|^2 \leq 2d(E^*, x^*) + \frac{1}{\sigma^2}.$$

Therefore,

$$D_{E,x} \stackrel{\text{def}}{=} \max_{E_i \in Q_k^{(i)}, x \in Q_x} d(E,x) \leq \left(\sqrt{d(E^*,x^*)} + \sqrt{d(E^*,x^*) + \frac{1}{2\sigma^2}} \right)^2$$
$$= 2d(E^*,x^*) + \frac{1}{2\sigma^2} + 2\sqrt{d(E^*,x^*)^2 + d(E^*,x^*)\frac{1}{2\sigma^2}}$$
$$\leq 4d(E^*,x^*) + \frac{1}{2\sigma^2} + \sqrt{2d(E^*,x^*)}\frac{1}{\sigma}.$$

By [15, Theorem 3], we further have

$$\delta_t \leq \hat{\beta}_{t+1}\left(D_{E,x}\sigma + \frac{1}{2\sigma}\right) \leq \hat{\beta}_{t+1}\left[4d(E^*,x^*)\sigma + \frac{1}{\sigma} + \sqrt{2d(E^*,x^*)}\right].$$

Minimizing the above last term in σ, we obtain that at $\sigma = 1/(2\sqrt{d(E^*,x^*)})$,

$$\delta_t \leq \hat{\beta}_{t+1}(4+\sqrt{2})\sqrt{d(E^*,x^*)}, \tag{22}$$
$$D_{E,x} \leq 2(3+\sqrt{2})d(E^*,x^*). \tag{23}$$

Let $\tau = \frac{1}{2}$. By (15), (21), and (23), we have

$$L_E^2 = \max_{E_i \in Q_k^{(i)}, x \in Q_x} \|g_E(E,x)\|_{F,*}^2 \leq mk + 8(3+\sqrt{2})\frac{\gamma}{r}L\|B\|_2^2 d(E^*,x^*). \tag{24}$$

Therefore,

$$\frac{1}{\sum_{l=1}^t \alpha_l} \leq \frac{1}{t+1}\sqrt{2L_E^2 + 2L_x^2}$$
$$\stackrel{(16),(24)}{\leq} \frac{1}{t+1}\Big[2mk + 16(3+\sqrt{2})\frac{\gamma}{r}L\|B\|_2^2 d(E^*,x^*)$$
$$+ 2\left(2\|f\|_2 + 2\sqrt{\gamma L(\rho_u - kr + r)}\|B\|_2\right)^2\Big]^{1/2}.$$

Along with (11) and (22), we obtain the duality gap

$$0 \leq \max_{x_j \in Q_x} F(\hat{E}^{(t+1)}, x) - \min_{E_i \in Q_k^{(i)}} F(E, \hat{x}^{(t+1)})$$
$$\leq \frac{(4\sqrt{2}+2)\hat{\beta}_{t+1}\sqrt{d(E^*,x^*)}}{t+1}\Big[mk + 8(3+\sqrt{2})\frac{\gamma}{r}L\|B\|_2^2 d(E^*,x^*)$$
$$+ 4\left(\|f\|_2 + \sqrt{\gamma L(\rho_u - kr + r)}\|B\|_2\right)^2\Big]^{1/2}.$$

2. *Bound* 2. Since

$$\alpha_l \geq \frac{1}{\sqrt{L_E^2/\tau + L_x^2/(1-\tau)}},$$

by [15, Theorem 3], we have

$$\begin{aligned} 0 \quad &\leq \quad \max_{x_j \in Q_x} F(\hat{E}^{(t+1)}, x) - \min_{E_i \in Q_k^{(i)}} F(E, \hat{x}^{(t+1)}) \ \leq \ \frac{\delta_t}{\sum_{l=0}^t \alpha_l} \\ &\leq \quad \frac{\hat{\beta}_{t+1}}{t+1} \left[\sigma(\tau D_E + (1-\tau) D_x) + \frac{1}{2\sigma} \right] \sqrt{L_E^2/\tau + L_x^2/(1-\tau)}. \end{aligned}$$

We choose

$$\sigma \ = \ \frac{1}{\sqrt{2\tau D_E + 2(1-\tau) D_x}}, \quad \tau \ = \ \frac{\sqrt{D_x} L_E}{\sqrt{D_E} L_x + \sqrt{D_x} L_E}.$$

Then

$$\begin{aligned} 0 \quad &\leq \quad \max_{x_j \in Q_x} F(\hat{E}^{(t+1)}, x) - \min_{E_i \in Q_k^{(i)}} F(E, \hat{x}^{(t+1)}) \\ &\leq \quad \frac{\sqrt{2} \hat{\beta}_{t+1}}{t+1} \left(L_E \sqrt{D_E} + L_x \sqrt{D_x} \right). \end{aligned}$$

From (9), (10), (15), (16), and (17), we have

$$\begin{aligned} 0 \quad &\leq \quad \max_{x_j \in Q_x} F(\hat{E}^{(t+1)}, x) - \min_{E_i \in Q_k^{(i)}} F(E, \hat{x}^{(t+1)}) \\ &\leq \quad \frac{0.37 + \sqrt{2t+1}}{t+1} \left[\sqrt{m^2 k + L^2 m \frac{\gamma}{r} \|B\|_2^2 \eta^2 (\rho_u - kr)} + 2\sqrt{L} \eta \|f\|_2 \right. \\ &+ \quad \left. 2\sqrt{\gamma(\rho_u - kr + r)} \|B\|_2 L \eta \right]. \end{aligned}$$ □

6.2. Computational cost of each iteration. The cost of each iteration of our algorithm has two components: that from calculating the subgradients and that from solving the subproblems.

1. *Cost of updating s^E and s^x.*

$$\begin{aligned} 0 \quad &\leq \quad \max_{x_j \in Q_x} F(\hat{E}^{(t+1)}, x) - \min_{E_i \in Q_k^{(i)}} F(E, \hat{x}^{(t+1)}) \\ &\leq \quad \frac{(4\sqrt{2}+2)\hat{\beta}_{t+1} \sqrt{d(E^*, x^*)}}{t+1} \left[mk + 8(3+\sqrt{2}) \frac{\gamma}{r} L \|B\|_2^2 d(E^*, x^*) \right. \\ &+ \quad \left. 4 \left(\|f\|_2 + \sqrt{\gamma L(\rho_u - kr + r)} \|B\|_2 \right)^2 \right]^{1/2}. \end{aligned}$$

We don't keep g_E and g_x in memory, but update s^E_{t+1} and s^x_{t+1} directly. Since g_E and g_x share some of the same components, we compute $\sum_{l=1}^{nig} B_{i,l} x_j x_j^T B_{i,l}$ and $A(E) x_j$ in the same loop. To balance the demands between memory and speed, we compute s^E_{t+1} and s^x_{t+1} as follows:

```
do j = 1 ... L
  if   j ∈ R
    0 → w
    α_t √γ ⟨ A(E)x_j , x_j ⟩^{-1/2} → u_j
    do i = 1 ... m
      Σ_{l=1}^{nig} B_{i,l} x_j x_j^T B_{i,l}^T → q
      w + A_i(E)x_j → w
      s^{E_i} − u_j q → s^{E_i}          (k(k+1) flops)
    end i
  else
    A(E)y → w                          (m·nig(4kN + 2k^2) flops)
  end if
  s^{x_j} + 2(α_t f_j − u_j w) → s^{x_j}   (5N flops)
end do j
s^E + α_t I_k → s^E                    (mk flops)
```

The inner products $\langle A(E)x_j, x_j\rangle$ are computed as follows:

```
0 → u
do i = 1 ... m
  do l = 1 ... nig
     B_{i,l} x_j → v          (2kN flops)
     E_i v → p                (2k^2 flops)
     u + v^T p → u            (2k flops)
  end do l
end do i
output s
```

In the algorithm, we keep the value $\sqrt{\gamma}$ in memory. Therefore, the arithmetic costs of calculating $\alpha_t\sqrt{\gamma}\langle A(E)x_j, x_j\rangle^{-1/2}$ for $j = 1,\dots,L$ are $\big(2L\cdot m\cdot nig\cdot[kN+k^2+k]+4L\big)$ flops. The total length of auxiliary vectors v, p, and u_j is $(N+k+L)$. After computing the u_j's, memory for v and p can be released. We compute $(\sum_{l=1}^{nig} B_{i,l}x_jx_j^T B_{i,l}^T)$ for g_E and $A(E)x_j$ for g_x in the same loop; i.e., the update of q and w in loop i of the above algorithm is done as follows:

```
q → 0
do l = 1 ... nig
  B_{i,l} x_j → v              (2kN flops)
  q + v v^T → q                (k(k+1) flops)
  E_i v → v                    (2k^2 flops)
  w + B_{i,l}^T v → w          (2kN flops)
end do l
```

The above `l` loop takes a total of $nig(4kN + 3k^2 + k)$ flops. It is executed at most Lm times. The total length of the auxiliary vectors v, q, and w is $(k + \frac{k(k+1)}{2} + N)$.

Adding everything together, we get that the total number of flops used in updating s^E and s^x is at most $(6kL \cdot nig)mN + [(5k^2 + 3k)L \cdot nig + (k^2 + k)L + k]m + (5L)N + 4L$, and at most $\left(\frac{1}{2}k^2 + \frac{3}{2}k + N + L\right)$ auxiliary storage space units are used.

2. *Cost of solving the subproblems.*

For $t = 0, \ldots$, from the closed-form solution (7) given in section 5, we obtain that it takes $L(3N + 7)$ flops to compute $x^{(t+1)}$. The value of $\beta_{t+1}\tau$ is stored for calculating $E^{(t+1)}$ later.

Now we consider the worst-case complexity of computing $E^{(t+1)}$. By the representation of $E^{(t+1)}$, it is obvious that the most computation is needed when

$$\lambda < 0, \qquad \sum_{i=1}^{k} \lambda_i < \beta_{t+1}\tau(kr - \rho_u).$$

Comparing $\sum_{q \in \bar{M}_0} \lambda_q$ with $\beta_{t+1}\tau(kr - \rho_u)$ and $\beta_{t+1}\tau(kr - \rho_l)$ takes $(2k + 7)$ flops and three auxiliary storage space units, since we can keep kr as an intermediate result. Similarly to the analysis in section A.2, we can obtain the complexity of Algorithm projSyml as follows: Step 1 takes at most $k(k-1)$ comparisons and exchanges. Because we have already calculated $\beta_{t+1}\tau(kr - \rho_u)$, two additions and subtractions are needed to obtain T. Step 2 takes at most $3(k-1)$ flops. Step 3 takes at most $(2 + 3k)$ steps. Therefore, a total of at most $(k^2 + 7k + 8)$ flops is needed to obtain ω, and $(k + 4)$ auxiliary space units are needed to store the sorted index set, T, $\beta_{t+1}\tau$, $\beta_{t+1}\tau(kr - \rho_u)$, q, since we overwrite the memory storing $\beta_{t+1}\tau(kr - \rho_l)$ by T.

Eigenvalue decomposition of $s_{t+1}^{E_i}$ takes about $9k^3$ flops and $k^2 + 2k + 1$ auxiliary storage space units. Computing $U \operatorname{diag}(\omega) U^T$ takes about $(k^2(k+1) + k^2)$ flops. Therefore, at most $m(10k^3 + 3k^2 + 7k + 8)$ flops and $(k^2 + 2k + 1)$ auxiliary storage space units are needed to obtain $E^{(t+1)}$.

For problem (2), k equals 3 or 6; L and nig are much smaller than m and N. After omitting small-order terms, we then conclude that about $(6kL \cdot nig)mN$ flops are needed for each iteration of our algorithm, and the auxiliary storage space units are about N.

On the other hand, to evaluate $\langle A(E)^{-1} f_j, f_j\rangle$ presented in the original formula (2), we need to first form the matrix $A(E)$, which requires $m \cdot nig \cdot [2k^2N + (k + \frac{1}{2})N(N + 1)]$ flops: computing $E_i B_{i,l}$ takes $2k^2N$ flops; calculating $B_{i,l}^T E_i B_{i,l}$ takes $kN(N + 1)$ flops; adding the $m \cdot nig$ matrices $B_{i,l}^T E_i B_{i,l}$ together requires $\frac{1}{2} m \cdot nig N(N + 1)$ flops. An auxiliary vector of size $\frac{N(N+1)}{2}$ is needed to store $A(E)$. We then compute the Cholesky factorization of $A(E) = CC^T$, which takes $\frac{N^3}{3}$ flops. Next we compute $z_j = C^{-T}(C^{-1}x_j)$ (for $j = 1, \ldots, L$), which needs $2LN^2$ flops. Finally, the inner products $\langle z_j, x_j\rangle$ take $2LN$ flops to compute. Therefore, a total of $m \cdot nig \cdot [2k^2N + (k + \frac{1}{2})N(N + 1)] + \frac{N^3}{3} + 2L(N^2 + N)$ flops and an auxiliary vector of size $\frac{N(N+1)}{2}$ are required to compute $\langle A(E)^{-1} f_j, f_j\rangle$ (for $j = 1, \ldots, L$). After omitting small-order terms, we conclude that about $\frac{1}{3}N^3$ flops and $\frac{1}{2}N^2$ auxiliary storage space units are needed to obtain $\langle A(E)^{-1} f_j, f_j\rangle$.

In summary, the number of flops and auxiliary storage space units per iteration of our algorithm are both one order smaller than that for evaluating $\langle A(E)^{-1} f_j, f_j\rangle$.

Furthermore, if the matrices $B_{i,l}$ are sparse, computational work per iteration and auxiliary storage space requirements of our algorithm will be even smaller.

7. Penalized Lagrangian. Because $\langle A(E)^{-1} f_j, f_j\rangle$ is convex in E and function $([\sqrt{a} - \gamma^{1/2}]_+)^2$ is convex and increasing in a, we conclude that $\left([\langle A(E)^{-1} f_j, f_j\rangle^{1/2} - \gamma^{1/2}]_+\right)^2$ is convex in E; see, for instance, [9, Proposition 2.1.8]. To have a faster rate of convergence to feasibility, we add to the objective of (2) a convex penalty function for the compliance constraint,

$$\sum_{j=1}^{L} \nu \left(\left[\langle A(E)^{-1} f_j, f_j\rangle^{1/2} - \gamma^{1/2} \right]_+ \right)^2,$$

where $\nu > 0$ is the penalty parameter.

Then the Lagrangian becomes

$$p(E,x) \stackrel{\text{def}}{=} F(E,x) + \sum_{j=1}^{L} \nu \left(\left[\langle A(E)^{-1} f_j, f_j\rangle^{1/2} - \gamma^{1/2} \right]_+ \right)^2,$$

which is convex in E and concave in x. A solution to

$$\min_{\substack{E_i \in Q_k^{(i)} \\ k=1,\dots,m}} \max_{\substack{x_j \in \mathbb{R}^N, \\ j=1,\dots,L}} p(E,x)$$

approximates that of model (2).

The gradient of $p(E,x)$ at (E,x) is

$$\begin{aligned} \nabla_{E_i} p(E,x) &= g_{E_i}(E,x) - \sum_{j \in W_E} \nu \left[1 - \gamma^{1/2} / \langle A(E)^{-1} f_j, f_j\rangle^{1/2} \right]_+ \\ &\quad \cdot \left(\sum_{l=1}^{nig} B_{i,l} A(E)^{-1} f_j f_j^T A(E)^{-1} B_{i,l}^T \right), \quad i = 1,\dots,m, \\ \nabla_{x_j} p(E,x) &= g_{x_j}(E,x), \quad j = 1,\dots,L. \end{aligned}$$

Similarly to Lemma 1, we have the following results about the bounded version of penalized Lagrangian method.

LEMMA 6. *Let $(\tilde{E}, \tilde{x})$ be a solution to*

$$\min_{\substack{E_i \in Q_k^{(i)} \\ i=1,\dots,m}} \max_{\substack{\|x_j\| \le \eta, \\ j=1,\dots,L}} p(E,x). \tag{25}$$

Let f^* be the optimal value of (2).

1. *If $\|\tilde{x}_j\| < \eta$ for $j = 1,\dots,L$, then $\tilde{E}$ is a solution to the original problem.*
2. *Otherwise, $\tilde{E}$ has the following properties:*
   (a) $F(\tilde{E}) \le F^*$.
   (b) $\sum_{j \in W_{\tilde{E}}} \left(\langle f_j, A(\tilde{E})^{-1} f_j\rangle^{1/2} - \gamma^{1/2} \right)$
   $\le 1 / \left[\sqrt{\frac{\nu}{(f^* - m\rho_l)|W_{\tilde{E}}|} + \frac{r^2 \lambda_{\min}^2(BB^T)\eta^2}{(f^* - m\rho_l)^2}} + \frac{r\lambda_{\min}(BB^T)\eta}{f^* - m\rho_l} \right]$.

Proof. The proof of item 1 is the same as that for Lemma 1, item 1. Item 2 can be proved similarly as Lemma 1, item 2. Below, we briefly give the proof.

For any fixed $E \in Q$, the point

$$x_j = \begin{cases} \frac{\eta}{\|A(E)^{-1} f_j\|} A(E)^{-1} f_j, & j \in W_E, \\ 0, & j \notin W_E \end{cases}$$

is feasible for

$$\max_{\substack{\|x_j\| \le \eta, \\ j=1,\ldots,L}} p(E, x)$$

with objective value

$$p_x(E) \stackrel{\mathrm{def}}{=} \langle I, E\rangle + \sum_{j \in W_E} \nu w_j^2(E) + 2\frac{\langle f_j, A(E)^{-1} f_j\rangle^{1/2}}{\|A(E)^{-1} f_j\|} \eta w_j(E),$$

where

$$w_j(E) \stackrel{\mathrm{def}}{=} \langle f_j, A(\tilde{E})^{-1} f_j\rangle^{1/2} - \gamma^{1/2}, \quad j \in W_E.$$

Therefore,

$$F^* - m\rho_l \ge \sum_{j \in W_{\tilde{E}}} \nu w_j^2(\hat{E}) + 2r\lambda_{\min}(BB^T)\eta w_j(\hat{E}),$$

from which we obtain

$$F^* - m\rho_l + |W_{\hat{E}}| r^2 \lambda_{\min}^2 (BB^T)\eta^2/\nu \ge \sum_{j \in W_{\tilde{E}}} \nu \left[w_j(\hat{E}) + r\lambda_{\min}(BB^T)\eta/\nu \right]^2$$

$$\ge \frac{\nu}{|W_{\hat{E}}|} \left[\sum_{j \in W_{\tilde{E}}} w_j(\hat{E}) + |W_{\hat{E}}| r\lambda_{\min}(BB^T)\eta/\nu \right]^2.$$

Hence,

$$\sum_{j \in W_{\tilde{E}}} w_j(\hat{E}) \le 1 \Big/ \left[\sqrt{\frac{\nu}{(f^* - m\rho_l)|W_{\hat{E}}|} + \frac{r^2 \lambda_{\min}^2 (BB^T)\eta^2}{(f^* - m\rho_l)^2}} + \frac{r\lambda_{\min}(BB^T)\eta}{F^* - m\rho_l} \right].$$

Observe that as $\eta \to +\infty$ and $\nu \to +\infty$, the set of saddle-points of (25) approaches that of (2).

We can apply the preceding algorithm to obtain a saddle-point of $p(E, x)$ as well, and the subproblems of this algorithm have closed-form solutions.

Bounds on duality gaps. To estimate the duality gap of each iteration, we first bound $\nabla_E p(E,x)$ as follows:

$$\begin{aligned}\|\nabla_E p(E,x)\|_{F,*} &\le L_E + \nu \left[\sum_{i=1}^{m} \operatorname{tr}\left(\sum_{j=1}^{L}\sum_{l=1}^{nig} B_{i,l}A(E)^{-1}f_j f_j^T A(E)^{-1}B_{i,l}^T\right)^2\right]^{1/2}\\ &\le L_E + \nu \operatorname{tr}\left(\sum_{i=1}^{m}\sum_{j=1}^{L}\sum_{l=1}^{nig} B_{i,l}A(E)^{-1}f_j f_j^T A(E)^{-1}B_{i,l}^T\right)\\ &= L_E + \nu \sum_{j=1}^{L}\sum_{i=1}^{m}\sum_{l=1}^{nig} f_j^T A(E)^{-1}B_{i,l}^T B_{i,l}A(E)^{-1}f_j\\ &\overset{\lambda_{\min}(E)=r}{\le} L_E + \nu/r \sum_{j=1}^{L} f_j^T A(E)^{-1} f_j\\ &\le L_E + \frac{\nu}{r^2\lambda_{\min}(B^TB)}\sum_{j=1}^{L}\|f_j\|_2^2,\end{aligned}$$

where the last inequality is from

$$\lambda_{\min}(A(E)) \ge \lambda_{\min}(E)\lambda_{\min}(B^TB) \;=\; r\lambda_{\min}(B^TB).$$

By (9), (18), (19), and (20) in section 6, we obtain that the duality gaps of the iterates are bounded as follows: For $t = 0, \dots,$

$$\begin{aligned}0 &\le \max_{x_j\in Q_x} F(\hat{E}^{(t+1)}, x) - \min_{E_i \in Q_k^{(i)}} F(E, \hat{x}^{(t+1)})\\ &\le gap + \frac{0.37+\sqrt{2t+1}}{t+1}\frac{\sqrt{m}(\rho_u - kr)\nu}{r^2\lambda_{\min}(B^TB)}\sum_{j=1}^{L}\|f_j\|_2^2.\end{aligned}$$

Cost of each iteration. Compared with (5), extra computation is needed to calculate $\langle A(E)^{-1}f_j, f_j\rangle$ for $j = 1,\dots,L$ in order to solve (25), which is $\mathcal{O}(N^3)$ flops; see the analysis at the end of section 6. Therefore, the total cost of each iteration for solving (25) is $\mathcal{O}(N^3)$ flops and $\mathcal{O}(N^2)$ memory space units.

8. Numerical examples. We present some computational examples which are done in the MATLAB environment on a Windows PC. For each run, the starting point is as follows: We choose E_0 to be the identity matrix with trace equal to the upper bound of trace. For $j = 1,\dots,L$, we let x_j be a vector with the same element and $\|x_j\|_2 = \eta$.

Figure 1 shows how the objective value and the violation of constraints vary with the number of iterations. The problem instance is `tc18_s1` from the academic test library of the Plato project (www.plato-n.org) with $m = 128$, $N = 298$, $L = 1$, and $nig = 4$.

The figure shows that during the first few iterations, the objective value decreases but the constraint-violation increases rapidly, where constraint-violation is measured by $\sum_{j=1}^{L}\min\left[(\langle A(E)^{-1}f_j, f_j\rangle - \gamma), 0\right]$. With iterations moving on, the constraint-violation decreases with the objective value.

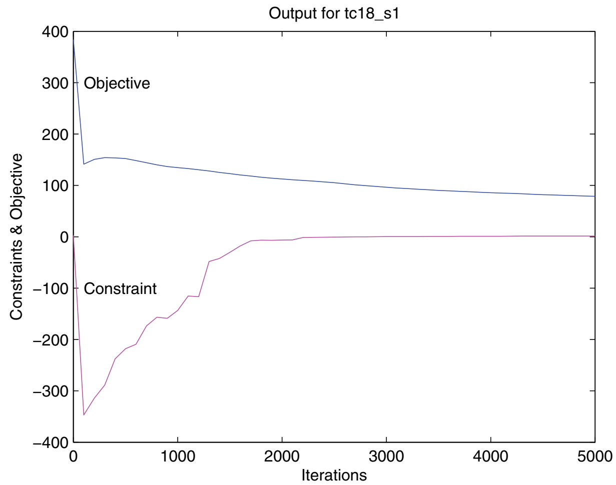

Fig. 1. *An example from the academic test library.*

In Tables 1 and 2, we present further numerical results on problems in the academic test library of the Plato project (www.plato-n.org). In the tables, column "cpu" give the total CPU times in seconds, column "obj" give the final objective values, column "obj-0" gives the initial objective values, and column "const" indicates whether the constraints are satisfied or not for the final solutions: "f" means "feasible." We compare formulas (5) and (25) on some infeasible problems, because constraints of these problems are difficult. The results are presented in Table 2. For each instance, we run 5000 gradient iterations. In Table 2, column "const" gives the sum of the values of the violation of constraints, i.e., $\sum_{j=1}^{L} \min\left[(\langle A(E)^{-1} f_j, f_j\rangle - \gamma), 0\right]$. Columns "obj-0" and "const-0" give objective values and the sum of values of the violation of constraints for the initial solutions. Columns "obj-p" and "const-p" give objective values and the sum of the values of the violation of constraints of the final solutions obtained by model (25). Columns "obj" and "const" give objective values and the sum of the values of the violation of constraints of the final solutions obtained by model (5).

From the results in Table 2, we see that the penalized Lagrangian can produce a better solution for infeasible problems, although it may not be the case for feasible problems. The penalty term forces iterates to move to the feasible region. On the other hand, because each iteration is much cheaper without calculating $\langle A(E)^{-1} f_j, f_j\rangle$, the penalized Lagrangian takes longer to solve a problem instance of FMO. The larger the dimension of the problem, the less time used by model (5) compared with model (25).

Appendix A. Matrix projection. Let $\mathcal{H}^n$ denote the space of $n\times n$ Hermitian matrices. We take the standard inner product on the space of complex square matrices of order n (or linear operators between Hilbert spaces of the same dimension): $\forall\, U, V \in \mathbb{C}^{n\times n}$,

$$\langle U, V\rangle \quad = \quad \mathrm{tr}(UV^*),$$

where V^* is the conjugate transpose of V. Let $\|\cdot\|_F$ denote the corresponding Frobenius norm. In this part, we give a closed-form solution to the following projection

TABLE 1
Examples on problems in academic test library.

Problem						Gradient method		
prob	m	N	L	nig	obj-0	cpu	obj	const
tc01_s1	96	216	1	4	288	2.77e+2	61.21	f
tc01_s2	384	816	1	4	1152	1.73e+3	8.82e+2	f
tc02_s1	96	216	1	4	288	2.96e+2	4.29	f
tc02_s2	384	816	1	4	1152	1.85e+3	6.43	f
tc03_s1	96	216	1	4	288	2.77e+2	60.12	f
tc03_s2	384	816	1	4	1152	1.68e+3	421.08	f
tc04_s1	300	670	1	4	900	1.26e+3	546.79	f
tc05_s1	800	1719	1	4	2.4e+3	5.19e+3	1.54e+3	f
tc07_s1	800	1680	1	4	2.4e+3	5.11e+3	6.75e+2	f
tc08_s1	128	272	1	4	384	3.79e+2	17.42	f
tc08_s2	512	1056	1	4	1536	2.48e+3	8.78e+2	f
tc14_s1	100	248	1	4	300	3.2e+2	51.98	f
tc14_s2	400	898	1	4	1200	1.8e+3	161.46	f
tc16_s1	128	300	1	4	384	3.9e+2	50.73	f
tc16_s2	512	1116	1	4	1536	2.74e+3	973.14	f
tc17_s1	128	300	1	4	384	4.14e+2	1.66e+2	f
tc17_s2	512	1116	1	4	1536	2.688e+3	5.54e+2	f
tc18_s1	128	298	1	4	384	4.0e+2	74.81	f
tc18_s2	512	1114	1	4	1536	2.57e+3	418.78	f
tc18sl_s1	128	298	1	4	384	3.85e+2	0.79	f
tc18sl_s2	512	1114	1	4	1536	2.57e+3	62.76	f
tc03_s1	96	216	2	4	288	4.24e+2	65.01	f
tc03_s2	384	816	2	4	1152	3.19e+3	739.23	f
tc06_s1	800	1719	3	4	2.4e+3	1.37e+4	1.5e+3	f
tc16_s1	128	300	2	4	384	5.91e+2	183.30	f
tc16_s2	512	1116	2	4	1536	4.32e+3	298.33	f
tc17_s1	128	300	2	4	384	5.99e+2	211.52	f
tc17_s2	512	1116	2	4	1536	4.04e+3	566.61	f
tc09_s1 (3d)	100	567	4	8	300	2.56e+3	73.95	f
tc09_s2 (3d)	512	2250	4	8	1536	3.66e+4	417.6	f
tc10_s1 (3d)	100	567	2	8	300	1.62e+3	51.71	f

problem:

$$\begin{aligned} \min_{Z\in\mathcal{H}^n} \quad & \|Z-U\|_F \\ \text{s.t.} \quad & c_l \le \operatorname{tr}(Z) \le c_u, \\ & \lambda_{\min}(Z) \ge r, \end{aligned} \tag{26}$$

where U is a square complex matrix of order n.

To this end, we first consider a least squares problem with nonnegativity constraint and a two-sided inequality.

A.1. Least squares with a two-sided inequality and nonnegative variables. Least squares problems have been studied intensively; however, we cannot find any reference for the problem discussed in this section. In this section, we first give an analytical solution of the problem; then we present an algorithm with total number of operations being a quadratic term in the dimension of problem variable.

Given $A \in \mathbb{R}^{n\times n}$ diagonal, $b \in \mathbb{R}^n$, $w \in \mathbb{R}^n$, $r \in \mathbb{R}^n$, $c_l \in \mathbb{R}\cup\{-\infty\}$, $c_u \in \mathbb{R}\cup\{+\infty\}$ with $c_l \le c_u$. Let$\|\cdot\|_2$ denote the norm induced by the inner product $\langle\cdot,\cdot\rangle$.

TABLE 2
Results with and without penalty function.

Problem							With penalty			Without penalty		
prob	m	N	L	nig	obj-0	const-0	cpu-p	obj-p	const-p	cpu	obj	const
bmat2x2 (1)	4	114	2	4	20	-10.99e+3	22.38	20	-6.87e+3	15.89	10.09	-1.7e+6
bmat2x2 (2)	4	114	2	4	12	-6.77e+6	22.55	12	-4.4e+6	15.25	5.53	-2.52e+9
bmat2x2 (3)	4	114	2	4	12	-8.75e+4	24.17	12	-1.02e+5	17.64	11.78	-8.03e+5
bmat1 (1)	16	40	1	4	80	-1.21e+4	63	79.93	-4.5e+3	41.89	20.09	-1.14e+7
bmat1 (2)	16	40	2	4	80	-3.17e+3	88.89	79.99	-3.39e+3	59.73	25.16	-1.25e+5
bmat1 (3)	16	40	2	4	48	-3.3e+6	86.58	47.99	-3.04e+6	59.66	36.29	-4.48e+6
bmat2 (1)	200	440	1	4	600	-1.98e+1	2.32e+4	573.07	-6.49	661.6	185.74	-6.44+3
bmat2 (2)	200	440	2	4	600	-3.02e+2	2.33e+4	303.04	-2.18e+2	1.03e+3	103.26	-2.82e+4
bmat2 (3)	200	440	2	4	1.0e+3	-7.3e+4	2.45e+4	785.73	-5.52e+4	1.03e+3	188.49	-1.49e+7
bmat (1)	400	850	1	4	2.0e+3	-9.11	1.64e+5	1991.12	-7.17	1.77e+3	794.46	-6.07e+3
bmat (2)	400	850	1	4	1.2e+3	-2.94e+2	1.66e+5	1175.92	-1.84e+2	2.79e+3	1121.55	-3.8e+2
bmat (3)	400	850	1	4	1.2e+3	-2.72e+2	1.66e+5	1187.93	-1.72e+2	3.14e+3	218.68	-4.18e+2
bmat1g	400	850	1	4	600	-1.22e+5	2.35e+4	488.4	-9.28e+4	1.07e+3	118.96	-5.78e+6

In this section, we give an analytical solution for the following least squares problem:

$$\begin{aligned} \min_{z\in\mathbb{R}^n} \quad & \|Az-b\|_2^2 \\ \text{s.t.} \quad & c_l \le \langle w, z\rangle \le c_u, \\ & z \ge r. \end{aligned} \tag{27}$$

Note that our problem includes the one-sided inequality case when $c_l = -\infty$ or $c_u = +\infty$, the lower bounded variable case when $c_l = -\infty$ and $c_u = +\infty$, and the equality case when $c_l = c_u$. Our problem also includes the case when not all variables are bounded, since we can replace an unconstrained variable $z_i \in \mathbb{R}$ by $z_i = z_i^+ - z_i^-$ with $z_i^+ \ge 0$, $z_i^- \ge 0$.

A.1.1. Problem reduction. To solve problem (27), we first show that we only need to consider the case with A being identity and $w_i \neq 0$ for $i = 1, \dots, n$.

If there exists ($\exists$) $a_{ii} = 0, w_i = 0$, we let

$$z_i^* = r_i.$$

If $\exists\, a_{ii} = 0, w_i > 0$, we let

$$z_i^* = \max\left\{ \left[c_l - \sum_{(1\le j\le n:\ a_{jj}\neq 0)} w_j \max\{b_j/a_{jj}, r_j\} \right]_+ \Big/ w_i, r_i \right\}.$$

If $\exists\, a_{ii} = 0, w_i < 0$, we let

$$z_i^* = \max\left\{ -\left[\sum_{(1\le j\le n:\ a_{jj}\neq 0)} w_j \max\{b_j/a_{jj}, r_j\} - c_u \right]_+ \Big/ w_i, r_i \right\}.$$

We also replace c_u and c_l by

$$c_u - w_i z_i^*, \quad c_l - w_i z_i^*.$$

If $\exists\, a_{ii} < 0$, we replace a_{ii} with $-a_{ii}$ and b_i with $-b_i$.

Hence after simplification, we can assume that A is a positive diagonal matrix in the text below. Thus, our least squares problem is equivalent to

$$\begin{aligned} \min_{z\in\mathbb{R}^n} \quad & \|z-b+Ar\|_2^2 \\ \text{s.t.} \quad & c_l - \langle w, r\rangle \le \langle A^{-1}w, z\rangle \le c_u - \langle w, r\rangle, \\ & z \ge 0. \end{aligned}$$

Therefore, for notational simplicity, we need only consider problem (27) in the following form:

$$\begin{aligned} \min_{z\in\mathbb{R}^n} \quad & \|z-b\|_2^2 \\ \text{s.t.} \quad & c_l \le \langle w, z\rangle \le c_u, \\ & z \ge 0. \end{aligned} \tag{28}$$

If $w_i = 0$ for some $i \in \{1, \dots, n\}$ in problem (28), then the corresponding solution of z_i must be $[b_i]_+$. After determining the solutions for these elements, we thereafter assume $w_i \neq 0$ for $i = 1, \dots, n$.

A.1.2. Analytical solution. In this section, we deduce the analytical solution for our least squares problem.

THEOREM 7. *The solution to* (28) *is*

$$z^* = \left[b - \frac{\left[\langle \tilde{w}, \tilde{b}\rangle - c_u\right]_+}{\|\tilde{w}\|_2^2} w + \frac{\left[c_l - \langle \tilde{w}, \tilde{b}\rangle\right]_+}{\|\tilde{w}\|_2^2} w \right]_+ ,$$

where $\tilde{w}$ *and* $\tilde{b}$ *denote the subvectors of* w *and* b *with indices in the set*

$$S = \left\{ 1 \le i \le n \colon b_i > \frac{\left[\langle \tilde{w}, \tilde{b}\rangle - c_u\right]_+}{\|\tilde{w}\|_2^2} w_i - \frac{\left[c_l - \langle \tilde{w}, \tilde{b}\rangle\right]_+}{\|\tilde{w}\|_2^2} w_i \right\}.$$

Proof. Because the constraints of problem (28) are linear, Lagrange multipliers exist. Let us write the Lagrangian function:

$$L(z, \lambda) = \|z - b\|_2^2 + \lambda_l \left(c_l - \langle w, z\rangle\right) + \lambda_u \left(\langle w, z\rangle - c_u\right), \quad (z \ge 0, \lambda_l \ge 0, \lambda_u \ge 0).$$

The solutions to problem (28) can be obtained by solving the following problem:

$$\max_{\lambda_l \ge 0, \lambda_u \ge 0} \min_{z \ge 0} L(z, \lambda).$$

Note that

$$L(z, \lambda) = \left\| z - b + \frac{\lambda_u - \lambda_l}{2} w \right\|_2^2 - \left(\frac{\lambda_u - \lambda_l}{2}\right)^2 \|w\|_2^2 + (\lambda_u - \lambda_l) \langle w, b\rangle + \lambda_l c_l - \lambda_u c_u,$$

from which we conclude that the solution to the Lagrangian dual $\min_{z \ge 0} L(z, \lambda)$ is

$$z^* = \left[b - \frac{\lambda_u - \lambda_l}{2} w \right]_+ .$$

We next determine the optimal values for λ_u and λ_l.

We first consider λ_l.

Let S denote the index set

$$S \stackrel{\text{def}}{=} \left\{ 1 \le i \le n \colon b_i > \frac{\lambda_u - \lambda_l}{2} w_i \right\}.$$

Let $\tilde{w}$ and $\tilde{b}$ denote the subvectors of w and b with indices in S. Let $\bar{b}$ denote the subvector of b with indices not in S. We then have

$$\begin{aligned} L(z^*, \lambda) &= -\left(\frac{\lambda_u - \lambda_l}{2}\right)^2 \|\tilde{w}\|_2^2 + (\lambda_u - \lambda_l) \langle \tilde{w}, \tilde{b}\rangle + \lambda_l c_l - \lambda_u c_u + \left\|\bar{b}\right\|_2^2 \\ &= -\left(\frac{\|\tilde{w}\|_2}{2} \lambda_l - \frac{\frac{\|\tilde{w}\|_2^2}{2} \lambda_u - \langle \tilde{w}, \tilde{b}\rangle + c_l}{\|\tilde{w}\|_2} \right)^2 + \left(\frac{\frac{\|\tilde{w}\|_2^2}{2} \lambda_u - \langle \tilde{w}, \tilde{b}\rangle + c_l}{\|\tilde{w}\|_2} \right)^2 \\ &\quad + \left\|\bar{b}\right\|_2^2 - \frac{\|\tilde{w}\|_2^2}{4} \lambda_u^2 + \lambda_u \langle \tilde{w}, \tilde{b}\rangle - \lambda_u c_u. \end{aligned}$$

Hence a solution of λ_l for $\max_{\lambda\geq 0} L(z^*, \lambda)$ must be in the form

$$\lambda_l^* \;=\; \left[\lambda_u - \frac{2}{\|\tilde{w}\|_2^2}\left(\langle \tilde{w}, \tilde{b}\rangle - c_l\right)\right]_+ .$$

To determine the solution of λ_u, we consider the following cases.

Case 1. For $\lambda_u < \frac{2}{\|\tilde{w}\|_2^2}(\langle \tilde{w}, \tilde{b}\rangle - c_l)$, the representation of λ_l^* is reduced to

$$\lambda_l^* \;=\; 0.$$

Since $\lambda_u \geq 0$, we have

$$\langle \tilde{w}, \tilde{b}\rangle \;>\; c_l.$$

Also,

$$\begin{aligned} L(z^*, \lambda_l^*, \lambda_u^*) &= -\frac{\|\tilde{w}\|_2^2}{4}\lambda_u^2 + \langle \tilde{w}, \tilde{b}\rangle\lambda_u - c_u\lambda_u + \|\bar{b}\|_2^2 \\ &= -\left(\frac{\|\tilde{w}\|_2}{2}\lambda_u - \frac{\langle \tilde{w}, \tilde{b}\rangle - c_u}{\|\tilde{w}\|_2}\right)^2 + \left\|\bar{b}\right\|_2^2 + \left(\frac{\langle \tilde{w}, \tilde{b}\rangle - c_u}{\|\tilde{w}\|_2}\right)^2 . \end{aligned}$$

Therefore, for this case, the solution to $\max_{\lambda\geq 0} L(z^*, \lambda)$ is

$$\lambda_u^* \;=\; \frac{2}{\|\tilde{w}\|_2^2}\left[\langle \tilde{w}, \tilde{b}\rangle - c_u\right]_+ .$$

Case 1.a. When $c_l < \langle \tilde{w}, \tilde{b}\rangle < c_u$, we have

$$\lambda_u^* \;=\; 0, \quad z^* \;=\; [b]_+.$$

Case 1.b. When $\langle \tilde{w}, \tilde{b}\rangle \geq c_u$, we have

$$\begin{aligned} \lambda_u^* &= \frac{2}{\|\tilde{w}\|_2^2}\left(\langle \tilde{w}, \tilde{b}\rangle - c_u\right), \\ z^* &= \left[b - \frac{\langle \tilde{w}, \tilde{b}\rangle - c_u}{\|\tilde{w}\|_2^2} w\right]_+ , \\ \langle w, z^*\rangle &= c_u. \end{aligned}$$

Case 2. For $\lambda_u \geq \frac{2}{\|\tilde{w}\|_2^2}(\langle \tilde{w}, \tilde{b}\rangle - c_l)$, we have

$$\lambda_l^* \;=\; \lambda_u - \frac{2}{\|\tilde{w}\|_2^2}\left(\langle \tilde{w}, \tilde{b}\rangle - c_l\right).$$

Also,

$$L(z^*, \lambda_l^*, \lambda_u) \;=\; (c_l - c_u)\,\lambda_u + \left(\frac{\langle \tilde{w}, \tilde{b}\rangle - c_l}{\|\tilde{w}\|_2}\right)^2 + \left\|\bar{b}\right\|_2^2 .$$

Therefore, in this case, the solution to $\max_{\lambda\geq 0} L(z^*,\lambda)$ is

$$\begin{aligned}
\lambda_u^* &= 0,\\
\lambda_l^* &= 2\frac{c_l - \langle \tilde{w}, \tilde{b}\rangle}{\|\tilde{w}\|_2^2},\\
z^* &= \left[b + \frac{c_l - \langle \tilde{w}, \tilde{b}\rangle}{\|\tilde{w}\|_2^2} w\right]_+,\\
\langle w, z^*\rangle &= c_l.
\end{aligned}$$

Because $\lambda_l^* \geq 0$, this case implies

$$\langle \tilde{w}, \tilde{b}\rangle \leq c_l.$$

Combining Cases 1 and 2, we obtain

$$z^* = \left[b - \frac{\left[\langle \tilde{w}, \tilde{b}\rangle - c_u\right]_+}{\|\tilde{w}\|_2^2} w + \frac{\left[c_l - \langle \tilde{w}, \tilde{b}\rangle\right]_+}{\|\tilde{w}\|_2^2} w\right]_+,$$

where $\tilde{w}$ denotes the subvector of w with indices in the set

$$S = \left\{1 \leq i \leq n \colon b_i > \frac{\left[\langle \tilde{w}, \tilde{b}\rangle - c_u\right]_+}{\|\tilde{w}\|_2^2} w_i - \frac{\left[c_l - \langle \tilde{w}, \tilde{b}\rangle\right]_+}{\|\tilde{w}\|_2^2} w_i\right\}.$$

Remark 8. In our deduction, it is obvious that for $c_l = -\infty$, we have $\lambda_l^* = 0$; for $c_u = +\infty$, we have $\lambda_u^* = 0$.

A.1.3. Algorithm. From the discussion in the previous section, we know that to find the optimal solution z^* of our least squares problem, we need only determine the set S. In this section, we describe how to find the set S for our solution.

Properties of S based on Lagrange multipliers. We first give some simple observations which will be used later on.

PROPOSITION 9. *Let $r_1 \in \mathbb{R}$, $r_3 \in \mathbb{R}$, $r_2 > 0$, $r_4 > 0$. Then*

$$\begin{aligned}
\frac{r_1}{r_2} > \frac{r_3}{r_4} \quad &\Leftrightarrow \frac{r_1}{r_2} > \frac{r_3 + r_1}{r_4 + r_2},\\
\frac{r_1}{r_2} < \frac{r_3}{r_4} \quad &\Leftrightarrow \frac{r_1}{r_2} < \frac{r_3 + r_1}{r_4 + r_2}.
\end{aligned}$$

We next give some properties of the set S based on Lagrange multipliers. Observe that both λ_l^* and λ_u^* cannot be positive at the same time. We organize our analysis based on scenarios depending on the signs of the Lagrange multipliers.

Case 1. $\lambda_u^* > 0$.

By the deduction above and Lagrange multiplier properties, we have the corresponding relations

$$\begin{aligned}\langle w, z\rangle &= c_u,\\ \langle \tilde{w}, \tilde{b}\rangle &> c_u,\\ \lambda_u^* &= 2\frac{\langle \tilde{w}, \tilde{b}\rangle - c_u}{\|\tilde{w}\|_2^2}.\end{aligned}$$

We next consider which indices are in the set S.

1. $S_1 \stackrel{\text{def}}{=} \{i\colon w_i > 0, b_i \geq 0\}$.

   LEMMA 10. *Suppose* $\frac{b_j}{w_j} \geq \frac{b_i}{w_i}$. *If* $i \in S$, *then* $j \in S$ *as well.*

   Proof. Assume $j \notin S$. Since $i \in S$, we have

   $$\frac{b_j w_j}{w_j^2} \geq \frac{b_i w_i}{w_i^2} > \frac{\langle \tilde{w}, \tilde{b}\rangle - c_u}{\|\tilde{w}\|_2^2}.$$

   By Proposition 9, we have

   $$\frac{b_j w_j}{w_j^2} > \frac{\langle \tilde{w}, \tilde{b}\rangle - c_u + b_j w_j}{\|\tilde{w}\|_2^2 + w_j^2}.$$

   Therefore, $j \in S$. ∎

2. $S_2 \stackrel{\text{def}}{=} \{i\colon w_i > 0, b_i < 0\}$.
   By the definition of S, we have $S_2 \nsubseteq S$.
3. $S_3 \stackrel{\text{def}}{=} \{i\colon w_i < 0, b_i \geq 0\}$.
   It is obvious $S_3 \subseteq S$.
4. $S_4 \stackrel{\text{def}}{=} \{i\colon w_i < 0, b_i < 0\}$.

   LEMMA 11. *Suppose* $\frac{b_j}{w_j} \leq \frac{b_i}{w_i}$. *If* $i \in S$, *then* $j \in S$ *as well.*

   Proof. Assume $j \notin S$. Since $i \in S$, we have

   $$\frac{b_j w_j}{w_j^2} \leq \frac{b_i w_i}{w_i^2} < \frac{\langle \tilde{w}, \tilde{b}\rangle - c_u}{\|\tilde{w}\|_2^2}.$$

   By Proposition 9, we have

   $$\frac{b_j w_j}{w_j^2} < \frac{\langle \tilde{w}, \tilde{b}\rangle - c_u + b_j w_j}{\|\tilde{w}\|_2^2 + w_j^2}.$$

   Therefore, $j \in S$. ∎

Case 2. $\lambda_l^* > 0$.
For this case, we have

$$\begin{aligned}\langle w, z^*\rangle &= c_l,\\ \langle \tilde{w}, \tilde{b}\rangle &< c_l,\\ \lambda_l^* &= 2\frac{c_l - \langle \tilde{w}, \tilde{b}\rangle}{\|\tilde{w}\|_2^2}.\end{aligned}$$

We now determine which indices are in the set S.

1. $S_1 \stackrel{\text{def}}{=} \{i\colon w_i > 0, b_i \geq 0\}$.
   By the definition of S, we have $S_1 \subseteq S$.
2. $S_2 \stackrel{\text{def}}{=} \{i\colon w_i > 0, b_i < 0\}$.
   Similar to the case for $\lambda_u^* > 0$, we have the following:
   Suppose $\frac{b_j}{w_j} \geq \frac{b_i}{w_i}$. If $i \in S$; then $j \in S$ as well.
3. $S_3 \stackrel{\text{def}}{=} \{i\colon w_i < 0, b_i \geq 0\}$.
   Similar to the case for $\lambda_u^* > 0$, we have the following:
   Suppose $\frac{b_j}{w_j} \leq \frac{b_i}{w_i}$. If $i \in S$; then $j \in S$ as well.
4. $S_4 \stackrel{\text{def}}{=} \{i\colon w_i < 0, b_i < 0\}$.
   By the definition of S, we have $S_4 \nsubseteq S$.

Case 3. $\lambda_l^* = \lambda_u^* = 0$.
For this case, we have

$$Z^* \;=\; [b]_+.$$

Determining the signs of Lagrange multipliers. We next show that whether the Lagrange multiplier is positive or not can be determined by $\langle w, [b]_+\rangle$.

LEMMA 12. *The Lagrange multiplier λ_l^* satisfies the following condition:*

$$\lambda_l^* \begin{cases} = 0 & \langle w, [b]_+\rangle \geq c_l, \\ > 0 & \langle w, [b]_+\rangle < c_l. \end{cases}$$

Proof. We first use contradiction to prove the result for the case $\langle w, [b]_+\rangle \geq c_l$. Assume $\lambda_l^* > 0$. By the properties for $\lambda_l^* > 0$, we have $c_l > \langle \tilde{w}, \tilde{b}\rangle$ and $S_1 \subseteq S$. Since $\langle w, [b]_+\rangle \geq c_l$, we must have $S_2 \cap S \neq \emptyset$.

Let $l \in S_2 \cap S$ such that $\frac{b_l}{w_l} \cdot \frac{w_j}{b_j} \geq 1$ $(\forall\, j \in S_2 \cap S)$. We would have

$$\sum_{j \in S_2 \cap S} \frac{b_l}{w_l} w_j^2 \;=\; \sum_{j \in S_2 \cap S} \left(\frac{b_l}{w_l} \cdot \frac{w_j}{b_j}\right) w_j b_j \;\;\leq\;\; \sum_{j \in S_2 \cap S} w_j b_j,$$

$$0 \;\;\leq\;\; \sum_{j \in S \setminus S_2} w_j b_j - c_l \;=\; \langle w, [b]_+\rangle - c_l.$$

Adding the above two inequalities together, we would have

$$-\frac{b_l}{w_l} \;\;\geq\;\; \frac{c_l - \sum_{j \in S} w_j b_j}{\sum_{j \in S_2 \cap S} w_j^2} \;\geq\; \frac{1}{2}\lambda_l^*,$$

contradicting $l \in S$.

We next consider the case $\langle w, [b]_+\rangle < c_l$.

By the assumption, we have

$$c_l - \langle w, [b]_+\rangle \;=\; c_l - \sum_{i \in S \cap (S_1 \cup S_3)} w_j b_j \;\;>\;\; 0$$

$$- \sum_{i \in S \cap S_2} w_j b_j \;\;\geq\;\; 0.$$

Adding the above two inequalities together, we have

$$\lambda_l^* \;>\; 0.$$

Similarly, we have the results for λ_u^*.

LEMMA 13. *The Lagrange multiplier λ_u^* satisfies the following condition:*

$$\lambda_u^* \begin{cases} = 0, & \langle w, [b]_+\rangle \le c_u, \\ > 0, & \langle w, [b]_+\rangle > c_u. \end{cases}$$

For the case $\lambda_u^* > 0$, deleting any index from the set $S_1 \cap S$ decreases the value $\frac{\langle \tilde{w}, \tilde{b}\rangle - c_u}{\|\tilde{w}\|_2^2}$, and deleting any index from the set $S_4 \cap S$ increases that value. Similarly, for the case $\lambda_l^* > 0$, deleting any index from the set $S_2 \cap S$ decreases the value $\frac{\langle \tilde{w}, \tilde{b}\rangle - c_l}{\|\tilde{w}\|_2^2}$, and deleting any index from the set $S_3 \cap S$ increases that value.

The discussion above proves that our algorithm below finds an optimal solution of the problem (27).

Algorithm. Reduce problem (27) to the form (28) and solve problem (28). Let n_i be the cardinality of the index set S_i, $(i = 1, \dots, 4)$. We first compute $\langle w, [b]_+\rangle$.

- If $\langle w, [b]_+\rangle \in [c_l, c_u]$, we let
$$z^* \;=\; [b]_+.$$
- If $\langle w, [b]_+\rangle > c_u$, we do the following.
  1. Reorder the elements in S_1 so that
  $$b_{\sigma(1)}/w_{\sigma(1)} \ge b_{\sigma(2)}/w_{\sigma(2)} \ge \cdots b_{\sigma(n_1)}/w_{\sigma(n_1)}.$$
  Reorder the elements in S_4 so that
  $$b_{\tau(1)}/w_{\tau(1)} \le b_{\tau(2)}/w_{\tau(2)} \le \cdots b_{\tau(n_1)}/w_{\tau(n_4)}.$$
  2. Let
  $$S = S_3, \quad T = \sum_{i \in S_3} w_i b_i - c_u, \quad v = \sum_{i \in S_3} w_i^2, \quad j = 1, \quad l = 1.$$
  3. Repeat the following two *while* loops until stable:
  While $v \frac{b_{\sigma(j)}}{w_{\sigma(j)}} > T$ and $j \le n_1$, do
  $$S \cup \{\sigma(j)\} \to S, \quad T + w_{\sigma(j)} b_{\sigma(j)} \to T, \quad v + w_{\sigma(j)}^2 \to v, \quad j+1 \to j.$$
  While $v \frac{b_{\tau(l)}}{w_{\tau(l)}} < T$ and $l \le n_4$, do
  $$S \cup \{\tau(l)\} \to S, \quad T + w_{\tau(l)} b_{\tau(l)} \to T, \quad v + w_{\tau(l)}^2 \to v, \quad l+1 \to l.$$
  4. Let
  $$z_i^* \;=\; \begin{cases} 0, & i \in \bar{S}, \\ b_i - \frac{T}{v} w_i, & i \in S. \end{cases}$$
- If $\langle w, [b]_+\rangle < c_l$, we do the following.
  1. Reorder the elements in S_2 so that
  $$b_{\sigma(1)}/w_{\sigma(1)} \ge b_{\sigma(2)}/w_{\sigma(2)} \ge \cdots b_{\sigma(n_2)}/w_{\sigma(n_2)}.$$
  Reorder the elements in S_3 so that
  $$b_{\tau(1)}/w_{\tau(1)} \le b_{\tau(2)}/w_{\tau(2)} \le \cdots b_{\tau(n_3)}/w_{\tau(n_3)}.$$

2. Let

$$S = S_1, \quad T = \sum_{i \in S_1} w_i b_i - c_l, \quad v = \sum_{i \in S_1} w_i^2, \quad j = 1, \quad l = 1.$$

3. Repeat the following two *while* loops until stable:
   (a) While $v \frac{b_{\sigma(j)}}{w_{\sigma(j)}} > T$ and $j \leq n_2$, let

$$S \cup \{\sigma(j)\} \to S, \quad T + w_{\sigma(j)} b_{\sigma(j)} \to T, \quad v + w_{\sigma(j)}^2 \to v, \quad j+1 \to j.$$

   (b) While $v \frac{b_{\tau(l)}}{w_{\tau(l)}} < T$ and $l \leq n_3$, let

$$S \cup \{\tau(l)\} \to S, \quad T + w_{\tau(l)} b_{\tau(l)} \to T, \quad v + w_{\tau(l)}^2 \to v, \quad l+1 \to l.$$

4. Let

$$z_i^* = \begin{cases} 0, & i \in \bar{S}, \\ b_i - \frac{T}{v} w_i, & i \in S. \end{cases}$$

LEMMA 14. *After reducing problem* (27) *to problem* (28)*, the algorithm above stops at an optimal solution to* (28) *with at most* $n^2 + 14n + 1$ *arithmetic operations and* $2n + 3$ *auxiliary storage space units. If all* $w_i = 1$*, the above algorithm needs at most* $n^2 + 7n + 1$ *arithmetic operations and* $n + 2$ *auxiliary storage space units.*

Proof. Determining the signs of b_i and computing $\langle w, [b]_+ \rangle$ takes $3n - 1$ flops. Further dividing the index set into $S_1, \ldots, S_4$ takes another n flops. Comparing $\langle w, [b]_+ \rangle$ with c_l and c_u takes two operations. Computing b_i/w_i $(i = 1, \ldots, n)$ takes n flops. Bubble sorting the elements in the sets $S_1, \ldots, S_4$ takes at most $n(n-1)$ operations. Two auxiliary vectors of size n are required to store b_j/w_j for $(j = 1, \ldots, n)$ and the sorted index set. The number of flops needed for steps 2 and 3 is at most $7n$. We also need three auxiliary space units to store j, v and T. Step 4 takes at most $3n$ flops. Since we overwrite b by z, we do not need an additional vector for z. Therefore, a total of $n^2 + 14n + 1$ operations and $2n + 3$ auxiliary storage space units are required for our algorithm. If all $w_i = 1$, we do not need to divide and multiply the intermediate results by w_j. The index sets S_3 and S_4 are not needed, and b_j/w_j does not need to be stored. Also, we do not need to keep and compute v, since its value equals j. Therefore, the total number of operations is reduced to at most $n^2 + 7n + 1$. ∎

A.2. Symmetric matrix projection with lower bounds and a two-sided linear constraint.

THEOREM 15. *For given* $U \in \mathbb{C}^n$*, and* $c_l, c_u, r \in \mathbb{R}$ *with* $c_u \geq \max\{nr, c_l\}$*, the solution* $\hat{Z}$ *to the projection problem* (26) *is the following.*

Let $Q\Lambda Q^*$ *be the eigenvalue decomposition of* $\frac{U+U^*}{2}$*. Let* λ *denote the diagonal entries of* Λ*.*

Denote

$$S_0 \stackrel{\text{def}}{=} \{1 \leq j \leq n \colon \lambda_j \leq r\}, \quad \bar{S}_0 \stackrel{\text{def}}{=} \{1, \ldots, n\} \setminus S_0.$$

1. *Assume* $c_l \leq \sum_{i \in \bar{S}_0} \lambda_i + |S_0| r \leq c_u$*.*
   Then we let

$$\hat{\omega}_i = \lambda_i \quad i \in \bar{S}_0, \qquad \hat{\omega}_i = r \quad i \in S_0.$$

2. *Assume* $\sum_{i\in\bar{S}_0}\lambda_i + |S_0|r > c_u$.
   Then there is a partition of $\bar{S}_0$ *as* $\bar{S}_0 = S \cup \bar{S}$*:*

$$S \overset{\text{def}}{=} \left\{ i \in \bar{S}_0 \colon \lambda_i > \frac{\sum_{j\in S}\lambda_j + nr - c_u}{|S|} \right\},$$

$$\bar{S} \overset{\text{def}}{=} \left\{ i \in \bar{S}_0 \colon \lambda_i \leq \frac{\sum_{j\in S}\lambda_j + nr - c_u}{|S|} \right\}.$$

   We let

$$\hat{\omega}_i = \begin{cases} r, & i \in \bar{S} \cup S_0, \\ \lambda_i - \frac{\sum_{j\in S}\lambda_j + nr - c_u}{|S|} + r, & i \in S. \end{cases}$$

3. *Assume* $\sum_{i\in\bar{S}_0}\lambda_i + |S_0|r < c_l$.
   Then there is a partition of S_0 *as* $S_0 = S_l \cup \bar{S}_l$ *where*

$$S_l = \left\{ i \in S_0 \colon \frac{c_l - \sum_{j\in S_l\cup\bar{S}_0}\lambda_j - nr}{|\bar{S}_0| + |S_l|} > -\lambda_i \right\}.$$

   We let

$$\hat{\omega}_i = \begin{cases} \lambda_i + \frac{c_l - \sum_{j\in\bar{S}_0\cup S_l}\lambda_j - |\bar{S}_l| r}{|\bar{S}_0| + |S_l|}, & i \in \bar{S}_0 \cup S_l, \\ r, & i \in \bar{S}_l. \end{cases}$$

Let $\hat{\Omega}$ *be the diagonal matrix with diagonal entries* $\hat{\omega}$*. Then* $\hat{Z} = Q\hat{\Omega}Q^*$ *is the unique solution to* (26).

If $U \in \mathcal{S}^n$*,* $\hat{Z}$ *can be obtained in* $(10n^3 + 3n^2 + 9n + 5)$ *flops with an auxiliary storage vector of size* $(n^2 + 3n + 4)$*.*

Proof. Since $Z \in \mathcal{H}^n$, we have

$$\begin{aligned} \|Z - U\|_F^2 &= \frac{1}{2}\left(\|Z - U\|_F^2 + \|Z - U^*\|_F^2\right) \\ &= \operatorname{tr}(Z^2) + \operatorname{tr}(UU^*) - \operatorname{tr}(ZU + ZU^*) \\ &= \operatorname{tr}\left(Z - \frac{U + U^*}{2}\right)^2 + \frac{1}{2}\operatorname{tr}(UU^*) - \frac{1}{4}\operatorname{tr}(U^2) - \frac{1}{4}\operatorname{tr}(U^{*2}). \end{aligned}$$

Therefore, the solution to (26) is the same as the solution to the following problem:

$$\begin{aligned} \min_{Z\in\mathcal{H}^n} \quad & \left\| Z - \frac{U + U^*}{2} \right\|_F^2 \\ \text{s.t.} \quad & c_l \leq \operatorname{tr}(Z) \leq c_u, \\ & \lambda_{\min}(Z) \geq r. \end{aligned}$$

Let $\hat{F}$ be the optimal value of the above problem.

By Theorem 7, $\hat{\omega}$ in the statement of the theorem is the solution to

$$\begin{aligned} \min_{\omega\geq r} \quad & \|\omega - \lambda\|_2 \\ \text{s.t.} \quad & c_l \leq \sum_{i=1}^n \omega_i \leq c_u. \end{aligned}$$

The Hoffman–Wielandt theorem [10] states that for two Hermitian matrices V and W, let $\lambda_1(V), \dots, \lambda_n(V)$ and $\lambda_1(W), \dots, \lambda_n(W)$ be the eigenvalues of V and W in nonincreasing order. Then there is a permutation $\sigma(i)$ $(i = 1, \dots, n)$ such that

$$\sum_{i=1}^{n} \left[\lambda_{\sigma(i)}(W) - \lambda_i(V)\right]^2 \quad = \quad \|W - V\|_F^2.$$

It is obvious from Theorem 7 that $\hat{\omega}$ is in the same order as λ; i.e., if λ is arranged in nonincreasing order, $\hat{\omega}$ is also in nonincreasing order. Therefore,

$$\hat{F} \quad \geq \quad \|\hat{\omega} - \lambda\|_2^2.$$

Since $\hat{Z}$ and $\frac{U+U^*}{2}$ are unitary similar, we have

$$\left\|\hat{Z} - \frac{U+U^*}{2}\right\|_F^2 \quad = \quad \|\hat{\omega} - \lambda\|_F^2.$$

Hence $\hat{Z}$ is the solution to (26).

Now we consider the complexity and memory requirement of getting the solution $\hat{Z}$ when U is real symmetric.

The eigenvalue decomposition of U by the symmetric QR algorithm takes roughly $9n^3$ flops. Since we can overwrite U, n^2 space units are needed to store the orthogonal matrix Q, and about $2n + 1$ auxiliary space units are needed to store intermediate results. The algorithm in section A.1.3 can be used to compute $\hat{\omega}$. Since all of the r_i's are identical, variable transformations from c_l and c_u to $\tilde{c}_l$ and $\tilde{c}_u$ take 4 flops, instead of $2n$ flops for r_i's being heterogeneous. Therefore, calculating $\hat{\omega}$ takes at most $(n^2 + 9n + 5)$ flops and $3n + 4$ auxiliary storage space units. Computing $Q\hat{\Omega}Q^*$ takes $(n^2(n+1)+n^2)$ flops. Since the auxiliary vector for storing the intermediate results of the eigenvalue decomposition of U can be overwritten, the total length of the auxiliary vectors is $(n^2 + 3n + 4)$, and the total number of flops is $(10n^3 + 3n^2 + 9n + 5)$ for $U \in \mathcal{S}^n$. ∎

If $n \leq 3$, the characteristic polynomial of $\frac{U+U^*}{2}$ is of order no more than 3; therefore, its eigenvalues can be obtained analytically. Its eigenvectors can then be obtained by solutions to its eigensystems.

Appendix B. Updating the parameters. As is stated earlier, by [15, Theorem 1], the duality gap of the tth iteration generated by the primal-dual algorithm is bounded by

$$\text{(29)} \qquad \frac{1}{\sum_{l=0}^{t} \alpha_t}\delta_t, \quad \text{with } \delta_t \;\leq\; \beta_{t+1}D + \frac{1}{2}\sum_{l=0}^{t}\frac{\alpha_l^2}{\beta_l}\|g_l\|_*^2.$$

In our algorithm, $\|g_l\|_* = \|[(g_E)_l, (g_x)_l]\|_*$, $D = \tau D_E + (1-\tau)D_x$.

For $t = 1, \dots,$ let

$$\text{(30)} \qquad \hat{\beta}_0 \;=\; \hat{\beta}_1 \;=\; 1, \quad \hat{\beta}_{t+1} \;=\; \hat{\beta}_t + \frac{1}{\hat{\beta}_t}.$$

Also,

$$\beta_t \;=\; \sigma\hat{\beta}_t.$$

Simple dual averages.

$$\alpha_t \;=\; 1.$$

Assume $\|g_t\|_* \le L$ for $t = 1, \ldots$; then by [15, Theorem 2], we have

$$\delta_t \;\le\; \hat{\beta}_{t+1}\left(D\sigma + \frac{1}{2\sigma}L^2\right), \quad \sum_{l=0}^{t}\alpha_l = t+1.$$

Weighted dual averages.

$$\alpha_t \;=\; \frac{1}{\|g_t\|_*}.$$

Assume $\|g_t\|_* \le L$ for $t = 1, \ldots$; then by [15, Theorem 3], we have

$$\delta_t \;\le\; \hat{\beta}_{t+1}\left(D\sigma + \frac{1}{2\sigma}\right), \quad \sum_{l=0}^{t}\alpha_l \;\ge\; \frac{t+1}{L}.$$

The above results show that the convergence rate of the algorithm depends on the choice of σ. It is not possible to determine the optimal σ without knowledge of D or L. In this part, we show how to dynamically update the parameter σ_t in the algorithm to obtain the best convergence rate.

Choosing β_t. Let $\sigma_0 > 0$ be the smallest possible value for σ. Let $w > 0$ be the number of steps for each test in updating σ.

The following algorithm is used for the choice of β_t.

1. Choose $w > 0, \sigma_0 > 0$.
2. Let
$$v \;=\; 0, \qquad \sigma \;=\; \sigma_0.$$
For $t = 0, \ldots, w$, let
$$\beta_t = \sigma_0\hat{\beta}_t.$$
3. Repeat the following until the convergence rate starts to decrease.
   - Let
   $$v \;=\; v+1, \qquad \sigma \;=\; 2*\sigma.$$
   - For $t = vw+1 \ldots (v+1)w$, let
   $$\beta_t \;=\; \sigma\hat{\beta}_t.$$
4. Let
$$v \;=\; v-1, \qquad \sigma \;=\; \sigma/2.$$
For $t = (v+2)w+1, \ldots,$ let
$$\beta_t \;=\; \sigma\hat{\beta}_t.$$

THEOREM 16. *The total number of test steps for the above procedure of determining σ is finite. The total number of iterations of the algorithm including the above procedure is at most 5/3 of the algorithm without the procedure but using optimal parameters plus a term in the order of $\mathcal{O}\left(\frac{1}{\epsilon}\right)$.*

Proof. Assume that at iteration t we obtained the σ from the above procedure. Set $v = v_t$. Suppose $\|g_l\|_* \le L$ $(l = 0, \dots, t)$. Since there is one backtrack period with w steps before landing at the current σ, from the above procedure, we have $\sigma = 2^{v_t} \cdot \sigma_0$ and $\beta_t = \sigma\hat{\beta}_t$.

To prove the theorem, we need to bound δ_t.

We first consider the *method of simple dual averages*. By (29),

$$\begin{aligned}
\delta_{t(s)} \quad &\le \quad \sigma_0 2^{v_t}\hat{\beta}_{t+1}D + \sum_{v=0}^{v_t+1} \frac{L^2}{\sigma_0 2^{v+1}} \sum_{l=v\cdot w+1}^{(v+1)w} \frac{1}{\hat{\beta}_l} + \sum_{l=(v_t+2)w+1}^{t} \frac{L^2}{2^{v_t+1}\sigma_0\hat{\beta}_l} \\
&= \sigma_0 2^{v_t}\hat{\beta}_{t+1}D + \frac{L^2}{\sigma_0 2^{v_t+1}} \sum_{l=0}^{t} \frac{1}{\hat{\beta}_l} + \frac{L^2}{\sigma_0 2^{v_t+1}} \sum_{v=1}^{v_t-1}\sum_{l=0}^{v\cdot w} \frac{1}{\hat{\beta}_l} - \frac{L^2}{\sigma_0 2^{v_t+2}} \sum_{l=(v_t+1)w+1}^{(v_t+2)w} \frac{1}{\hat{\beta}_l} \\
&\overset{(30)}{=} \hat{\beta}_{t+1}\left(\sigma_0 2^{v_t}D + \frac{1}{\sigma_0 2^{v_t+1}}L^2\right) + \frac{L^2}{\sigma_0 2^{v_t+1}} \sum_{v=1}^{v_t-1} \hat{\beta}_{v\cdot w+1} \\
&\quad + \frac{L^2}{\sigma_0 2^{v_t+2}}\left[\hat{\beta}_{(v_t+1)w+1} - \hat{\beta}_{(v_t+2)w+1}\right].
\end{aligned}$$

To further estimate the bound, we use [15, Lemma 3]:

$$\hat{\beta}_t \;\le\; \frac{1}{1+\sqrt{3}} + \sqrt{2t-1}, \quad t \ge 1.$$

From the above result, we have

$$\begin{aligned}
\sum_{v=1}^{v_t-1} \hat{\beta}_{v\cdot w+1} \quad &\le \quad \frac{v_t-1}{1+\sqrt{3}} + \sum_{v=1}^{v_t-1}\sqrt{2vw+1} \\
&\le \quad \frac{v_t-1}{1+\sqrt{3}} + \sqrt{\frac{1}{v_t-1}\sum_{v=1}^{v_t-1}(2vw+1)} \\
&= \quad \frac{v_t-1}{1+\sqrt{3}} + \sqrt{v_t w + 1} \\
&\le \quad 2^{v_t}\sqrt{w/2}.
\end{aligned}$$

The optimal value of σ is $\sigma^* = \frac{L}{\sqrt{2D}}$. The total number of iterations decreases with σ for $\sigma < \sigma^*$ and increases with σ for $\sigma > \sigma^*$. Therefore, we have

$$\begin{aligned}
v_t \quad &\le \quad \frac{1}{2} + \log_2\frac{L}{\sigma_0\sqrt{D}}, \\
\tfrac{\sigma^*}{2} \;\le\; \sigma \quad &\le \quad 2\sigma^*.
\end{aligned}$$

From the above inequalities, we obtain that the total number of test steps for the *method of simple dual averages* to obtain an optimal σ is no more than $\lceil \frac{5}{2} + \log_2\frac{L}{\sigma_0\sqrt{D}} \rceil w$. Also, $(D\sigma + \frac{1}{2\sigma}L^2)/(D\sigma^* + \frac{1}{2\sigma^*}L^2) \le 5/3$. Therefore, the total number of iterations of our procedure for the *method of simple dual averages* is at most 5/3 of that with optimal parameter plus $\mathcal{O}\big(\frac{\sqrt{w}L^2}{2\sqrt{2}\sigma_0\epsilon}\big)$.

Similarly, for the *method of weighted dual averages*, we have

$$\delta_{t(d)} \quad \leq \quad \hat{\beta}_{t+1}\left(\sigma_0 2^{v_t} D + \frac{1}{\sigma_0 2^{v_t+1}}\right) + \frac{\sqrt{w}}{\sigma_0 2\sqrt{2}}.$$

The optimal value of σ is $\sigma^* = \frac{1}{\sqrt{2D}}$. Therefore, we obtain

$$v_t \quad \leq \quad \frac{1}{2} - \log_2 \sigma_0 \sqrt{D},$$

$$\frac{\sigma^*}{2} \leq \sigma \quad \leq \quad 2\sigma^*.$$

Since

$$\sum_{l=0}^{t} \geq \frac{t+1}{L},$$

we conclude that the total number of test steps needed for the *method of weighted dual averages* to obtain an optimal σ is no more than $\lceil \frac{5}{2} - \log_2 \sigma_0 \sqrt{D} \rceil w$. Also, $(D\sigma + \frac{1}{2\sigma})/(D\sigma^* + \frac{1}{2\sigma^*}) \leq 5/3$. Therefore, the total number of iterations of our procedure for the *method of weighted dual averages* is at most 5/3 of those done by the original algorithm with optimal parameter plus $\mathcal{O}\big(\frac{\sqrt{w}L}{2\sqrt{2}\sigma_0\epsilon}\big)$. ∎

The worst-case complexity bound of the original algorithm is $\mathcal{O}(\frac{1}{\epsilon^2})$ [15]. Since our procedure adds a term of $\mathcal{O}(\frac{1}{\epsilon})$, the complexity remains at $\mathcal{O}(\frac{1}{\epsilon^2})$.

Acknowledgments. We thank the associate editor and anonymous referees for their helpful comments and suggestions.

REFERENCES

[1] K. J. Arrow, L. Hurwicz, and H. Uzawa, *Studies in Linear and Non-linear Programming*, with contributions by H. B. Chenery, S. M. Johnson, S. Karlin, T. Marschak, and R. M. Solow, Stanford Math. Stud. Social Sci. II, Stanford University Press, Stanford, CA, 1958.

[2] A. Ben-Tal, M. Kočvara, A. Nemirovski, and J. Zowe, *Free material design via semidefinite programming: The multiload case with contact conditions*, SIAM J. Optim., 9 (1999), pp. 813–832, https://doi.org/10.1137/S1052623497327994.

[3] M. P. Bendsøe, J. M. Guedes, R. B. Haber, P. Pedersen, and J. E. Taylor, *An analytical model to predict optimal material properties in the context of optimal structural design*, Trans. ASME J. Appl. Mech., 61 (1994), pp. 930–937, https://doi.org/10.1115/1.2901581.

[4] A. Chambolle and T. Pock, *A first-order primal-dual algorithm for convex problems with applications to imaging*, J. Math. Imaging Vision, 40 (2011), pp. 120–145, https://doi.org/10.1007/s10851-010-0251-1.

[5] S. Czarnecki and T. Lewiński, *A stress-based formulation of the free material design problem with the trace constraint and multiple load conditions*, Structural Multidiscip. Optim., 49 (2014), pp. 707–731, https://doi.org/10.1007/s00158-013-1023-5.

[6] O. Devolder, F. Glineur, and Y. Nesterov, *First-order methods of smooth convex optimization with inexact oracle*, Math. Program., 146 (2014), pp. 37–75, https://doi.org/10.1007/s10107-013-0677-5.

[7] I. Ekeland and R. Téman, *Convex Analysis and Variational Problems*, Classics Appl. Math. 28, SIAM, Philadelphia, 1999, https://doi.org/10.1137/1.9781611971088.

[8] J. Haslinger, M. Kočvara, G. Leugering, and M. Stingl, *Multidisciplinary free material optimization*, SIAM J. Appl. Math., 70 (2010), pp. 2709–2728, https://doi.org/10.1137/090774446.

[9] J.-B. Hiriart-Urruty and C. Lemaréchal, *Convex Analysis and Minimization Algorithms. I. Fundamentals*, Grundlehren Math. Wissen. 305 [Fundamental Principles of Mathematical Sciences], Springer-Verlag, Berlin, 1993, https://doi.org/10.1007/978-3-662-02796-7.
[10] A. J. Hoffman and H. W. Wielandt, *The variation of the spectrum of a normal matrix*, Duke Math. J., 20 (1953), pp. 37–39.
[11] M. Kočvara, M. Stingl, and J. Zowe, *Free material optimization: Recent progress*, Optimization, 57 (2008), pp. 79–100, https://doi.org/10.1080/02331930701778908.
[12] A. Nemirovski, *Prox-method with rate of convergence $O(1/t)$ for variational inequalities with Lipschitz continuous monotone operators and smooth convex-concave saddle point problems*, SIAM J. Optim., 15 (2004), pp. 229–251, https://doi.org/10.1137/S1052623403425629.
[13] Y. Nesterov, *Smooth minimization of non-smooth functions*, Math. Program., 103 (2005), pp. 127–152, https://doi.org/10.1007/s10107-004-0552-5.
[14] Y. Nesterov, *Dual extrapolation and its applications to solving variational inequalities and related problems*, Math. Program., 109 (2007), pp. 319–344, https://doi.org/10.1007/s10107-006-0034-z.
[15] Y. Nesterov, *Primal-dual subgradient methods for convex problems*, Math. Program., 120 (2009), pp. 221–259, https://doi.org/10.1007/s10107-007-0149-x.
[16] Y. Nesterov, *Barrier subgradient method*, Math. Program., 127 (2011), pp. 31–56, https://doi.org/10.1007/s10107-010-0421-3.
[17] Y. Nesterov, *Gradient methods for minimizing composite functions*, Math. Program., 140 (2013), pp. 125–161, https://doi.org/10.1007/s10107-012-0629-5.
[18] Y. Nesterov, *Universal gradient methods for convex optimization problems*, Math. Program., 152 (2015), pp. 381–404, https://doi.org/10.1007/s10107-014-0790-0.
[19] U. Ringertz, *On finding the optimal distribution of material properties*, Structural Optim., 5 (1993), pp. 265–267, https://doi.org/10.1007/BF01743590.
[20] M. Stingl, M. Kočvara, and G. Leugering, *Free material optimization with fundamental eigenfrequency constraints*, SIAM J. Optim., 20 (2009), pp. 524–547, https://doi.org/10.1137/080717122.
[21] M. Stingl, M. Kočvara, and G. Leugering, *A sequential convex semidefinite programming algorithm with an application to multiple-load free material optimization*, SIAM J. Optim., 20 (2009), pp. 130–155, https://doi.org/10.1137/070711281.
[22] A. G. Weldeyesus and M. Stolpe, *A primal-dual interior point method for large-scale free material optimization*, Comput. Optim. Appl., 61 (2015), pp. 409–435, https://doi.org/10.1007/s10589-014-9720-6.
[23] Y. Xia, *Gradient methods and conic least-squares problems*, Optim. Methods Software, 30 (2015), pp. 769–803.
[24] J. Zowe, M. Kočvara, and M. P. Bendsøe, *Free material optimization via mathematical programming*, Math. Program., 79 (1997), pp. 445–466, https://doi.org/10.1007/BF02614328.