ON THE MINIMUM VOLUME COVERING ELLIPSOID OF ELLIPSOIDS*

E. ALPER YILDIRIM†

Abstract. Let $\mathcal{S}$ denote the convex hull of m full-dimensional ellipsoids in $\mathbb{R}^n$. Given $\epsilon > 0$ and $\delta > 0$, we study the problems of computing a $(1 + \epsilon)$-approximation to the minimum volume covering ellipsoid of $\mathcal{S}$ and a $(1 + \delta)n$-rounding of $\mathcal{S}$. We extend the first-order algorithm of Kumar and Yıldırım [*J. Optim. Theory Appl.*, 126 (2005), pp. 1–21] that computes an approximation to the minimum volume covering ellipsoid of a finite set of points in $\mathbb{R}^n$, which, in turn, is a modification of Khachiyan's algorithm [L. G. Khachiyan, *Math. Oper. Res.*, 21 (1996), pp. 307–320]. Our algorithm can also compute a $(1 + \delta)n$-rounding of $\mathcal{S}$. For fixed $\epsilon > 0$ and $\delta > 0$, we establish polynomial-time complexity results for the respective problems, each of which is linear in the number of ellipsoids m. In particular, our algorithm can approximate the minimum volume covering ellipsoid of $\mathcal{S}$ in asymptotically the same number of iterations as that required by the algorithm of Kumar and Yıldırım to approximate the minimum volume covering ellipsoid of a set of m points. The main ingredient in our analysis is the extension of polynomial-time complexity of certain subroutines in the algorithm from a set of points to a set of ellipsoids. As a byproduct, our algorithm returns a finite "core" set $\mathcal{X} \subseteq \mathcal{S}$ with the property that the minimum volume covering ellipsoid of $\mathcal{X}$ provides a good approximation to the minimum volume covering ellipsoid of $\mathcal{S}$. Furthermore, the size of the core set depends only on the dimension n and the approximation parameter ϵ, but not on the number of ellipsoids m. We also discuss the extent to which our algorithm can be used to compute an approximate minimum volume covering ellipsoid and an approximate n-rounding of the convex hull of other sets in $\mathbb{R}^n$. We adopt the real number model of computation in our analysis.

Key words. minimum volume covering ellipsoids, Löwner ellipsoids, core sets, rounding of convex sets, approximation algorithms

AMS subject classifications. 90C25, 65K05, 90C22

DOI. 10.1137/050622560

1. Introduction. Given m full-dimensional ellipsoids $\mathcal{E}_1, \ldots, \mathcal{E}_m$ in $\mathbb{R}^n$, let $\mathcal{S}$ denote their convex hull. In this paper, we are concerned with the problems of approximating the minimum volume covering ellipsoid (MVCE) of $\mathcal{S}$, denoted by MVCE($\mathcal{S}$), also known as the Löwner ellipsoid of $\mathcal{S}$, and computing an approximate n-rounding of $\mathcal{S}$.

Ellipsoidal approximations of a compact convex set $\mathcal{S} \subset \mathbb{R}^n$ with a nonempty interior play an important role in several applications. By the Löwner–John theorem (see Theorem 2.1), MVCE($\mathcal{S}$) provides a good rounding of the set $\mathcal{S}$, which implies that certain characteristics of $\mathcal{S}$ can be approximated using an ellipsoidal rounding as long as MVCE($\mathcal{S}$) can be computed efficiently. For instance, an ellipsoidal rounding of $\mathcal{S}$ can be used to efficiently compute lower and upper bounds for a quadratic optimization problem over $\mathcal{S}$ (see Proposition 2.6).

The idea of approximating complicated objects using simpler ones is widely used in computational geometry and computer graphics. A common approach is to replace a complicated but more realistic model of a complex object with a simpler model of a less complex object covering the original object such as a minimum volume box

---

or a sphere. More recently, ellipsoidal models have been proposed in the literature as they usually provide better approximations than bounding boxes or spheres (see, e.g., [25, 26, 14, 10]). The key idea is to construct a so-called bounding volume hierarchy [11], which is simply a tree of bounding volumes. The bounding volume at a given node encloses the bounding volumes of its children. The bounding volume of a leaf encloses a primitive. Such a data structure can be used for detection collision or ray tracing. For instance, if a ray misses the bounding volume of a particular node, then the ray will miss all of its children, and the children can be skipped. The ray tracing algorithm traverses this hierarchy, usually in depth-first order, and determines if the ray intersects an object. Therefore, if an ellipsoidal approximation is used, the construction of a bounding volume hierarchy requires the computation of the MVCE of a union of ellipsoids at every node.

There is an extensive body of research on MVCEs of a finite set of points. We refer the reader to [15, 29, 18] and the references therein for a detailed account of numerous applications and several algorithms. In contrast, we are not aware of any specialized algorithms for the MVCE of ellipsoids in the literature. It is known that the problem can be formulated as an instance of a convex determinant optimization problem with linear matrix inequalities [5, 2, 6], which is amenable to theoretically efficient algorithms proposed in [32, 31]. Our main objective in this paper is to establish that the problem of MVCE of ellipsoids admits a sufficiently rich structure that enables us to extend the first-order algorithm of Kumar and Yıldırım [18], which, in turn, is a modification of Khachiyan's algorithm [15], that computes an approximate MVCE of a finite set of points in an almost verbatim fashion to a set of ellipsoids. The main ingredient in our analysis is the extension of polynomial-time complexity of certain subroutines in the algorithm of [18] from a set of points to a set of ellipsoids. We mainly rely on the complexity results of Porkolab and Khachiyan [21] on semidefinite optimization with a fixed number of constraints, which leads to the polynomial-time complexity of quadratic optimization over an ellipsoid—one of the subroutines in our algorithm (see Proposition 2.6). Throughout this paper, we adopt the real number model of computation [4]; i.e., arithmetic operations with real numbers and comparisons can be done with unit cost.

Given $\epsilon > 0$ and a compact convex set $\mathcal{S} \subset \mathbb{R}^n$, an ellipsoid $\mathcal{E}$ is said to be a $(1 + \epsilon)$-approximation to $\mathrm{MVCE}(\mathcal{S})$ if

$$(1) \qquad\qquad \mathcal{E} \supseteq \mathcal{S}, \quad \mathrm{vol}\,\mathcal{E} \leq (1 + \epsilon)\,\mathrm{vol}\,\mathrm{MVCE}(\mathcal{S}),$$

where $\mathrm{vol}\,\mathcal{E}$ denotes the volume of $\mathcal{E}$. Given $\delta > 0$ and a compact convex set $\mathcal{S} \subset \mathbb{R}^n$, an ellipsoid $\tilde{\mathcal{E}}$ is said to be a $(1 + \delta)n$-rounding of $\mathcal{S}$ if

$$(2) \qquad\qquad \frac{1}{(1 + \delta)n} \tilde{\mathcal{E}} \subseteq \mathcal{S} \subseteq \tilde{\mathcal{E}},$$

where the ellipsoid on the left-hand side of (2) is obtained by scaling $\tilde{\mathcal{E}}$ around its center by a factor of $1/((1+\delta)n)$. If $\mathcal{S}$ is centrally symmetric (i.e., $\mathcal{S} = -\mathcal{S}$), then we replace the factor on the left-hand side by $1/\sqrt{(1 + \delta)n}$. In this paper, we extend the first-order algorithm of [18] to compute a $(1+\epsilon)$-approximation to the MVCE of ellipsoids for $\epsilon > 0$. In particular, we establish that our extension has precisely the same iteration complexity as that of the algorithm of [18] (see Theorem 4.7). Furthermore, the overall complexity result is given by $O(mn^{O(1)}(\log n + [(1 + \epsilon)^{2/n} - 1]^{-1}))$, which depends only linearly on the number of ellipsoids m (see Theorem 4.8). In addition, our algorithm can also compute a $(1 + \delta)n$-rounding of the convex hull of a finite

number of ellipsoids for $\delta > 0$ in $O(mn^{O(1)}(\log n + \delta^{-1}))$ arithmetic operations (see Corollary 5.1). In both complexity results, $O(1)$ denotes a universal constant greater than four that does not depend on the particular instance. Therefore, our algorithm has polynomial-time complexity for fixed $\epsilon > 0$ and for fixed $\delta > 0$ and is especially well-suited for instances with $m \gg n$ and moderately small values of ϵ or δ.

As a byproduct, our algorithm computes a finite set $\mathcal{X} \subset \cup_{i=1,...,m}\mathcal{E}_i$ with the property that the convex hull of $\mathcal{X}$, denoted by conv$(\mathcal{X})$, provides a good approximation of $\mathcal{S} = \text{conv}\left(\cup_{i=1,...,m}\mathcal{E}_i\right)$. Moreover, the size of $\mathcal{X}$ depends only on the dimension n and the parameter ϵ but is independent of the number of ellipsoids m. In particular, $\mathcal{X}$ satisfies

$$\text{vol MVCE}(\mathcal{X}) \leq \text{vol MVCE}(\mathcal{S}) \leq \text{vol } \mathcal{E} \leq (1 + \epsilon)\text{vol MVCE}(\mathcal{X}) \leq (1 + \epsilon)\text{vol MVCE}(\mathcal{S}),$$

where $\mathcal{E}$ denotes the $(1 + \epsilon)$-approximation to the MVCE of $\mathcal{S}$ computed by our algorithm, which implies that $\mathcal{E}$ is simultaneously a $(1 + \epsilon)$-approximation to MVCE$(\mathcal{X})$ and to MVCE$(\mathcal{S})$ (see Proposition 4.9).

Following the literature, we refer to $\mathcal{X}$ as an "ϵ-core set" (or a "core set") [8, 7, 17, 18] since conv$(\mathcal{X})$ provides a compact approximation to the input set $\mathcal{S}$. Recently, core sets have received significant attention, and small core set results have been established for several geometric optimization problems such as the minimum enclosing ball problem and related clustering problems [17, 8, 7, 9, 1, 18]. Small core set results form a basis for developing practical algorithms for large-scale problems since many geometric optimization problems can be solved efficiently for small input sets.

The paper is organized as follows. We define our notation in the remainder of this section. In section 2, we present some preliminary results and discuss the complexity of semidefinite feasibility and optimization. We then establish that the ellipsoid containment problem can be cast as a linear matrix inequality and can therefore be checked in polynomial time. Section 3 is devoted to a deterministic volume approximation algorithm that will serve as an initialization stage for our algorithm. In section 4, we present and analyze a first-order algorithm for the MVCE problem. Section 5 establishes that our algorithm can also be used to compute an approximate n-rounding. We discuss how to extend our algorithm to other input sets in section 6. Section 7 concludes the paper with future research directions.

1.1. Notation. Vectors will be denoted by lowercase roman letters. For a vector u, u_i denotes its ith component. Inequalities on vectors will apply to each component. e will be reserved for the vector of ones in the appropriate dimension, which will be clear from the context. e^j is the jth unit vector. Uppercase roman letters will be reserved for matrices. $\mathcal{S}^n$ denotes the space of $n \times n$ real symmetric matrices. The inner product in $\mathcal{S}^n$ is given by $U \bullet V := \text{trace}(UV) = \sum_{i,j} U_{ij}V_{ij}$ for any $U, V \in \mathcal{S}^n$. Note that $u^T A u = A \bullet uu^T$ for any $A \in \mathcal{S}^n$ and $u \in \mathbb{R}^n$. For $A \in \mathcal{S}^n$, $A \succ 0$ $(A \succeq 0)$ indicates that A is positive definite (semidefinite) (i.e., the eigenvalues of A are strictly positive (nonnegative)). $\det(A)$ and rank(A) denote the determinant and the rank of a square matrix A, respectively. The identity matrix will be denoted by I. For a finite set of vectors $\mathcal{V}$, span$(\mathcal{V})$ denotes the linear subspace spanned by the vectors in $\mathcal{V}$. The convex hull of a set $\mathcal{T} \in \mathbb{R}^n$ is referred to as conv$(\mathcal{T})$. For a function $f : \mathbb{R}^n \to \mathbb{R}$, we use $x^* = \arg\max f(x)$ and $x_* = \arg\min f(x)$ to denote a global maximizer and a global minimizer of f, respectively. Superscripts will be used to refer to members of a sequence of vectors or matrices. Lowercase Greek letters will represent scalars. i, j, and k will be reserved for indexing purposes, and m and n will refer to the problem

data. Uppercase calligraphic letters will be used for all other objects such as sets, operators, and ellipsoids.

2. Preliminaries. A full-dimensional ellipsoid $\mathcal{E}$ in $\mathbb{R}^n$ admits a representation that is specified by an $n \times n$ symmetric positive definite matrix Q and a center $c \in \mathbb{R}^n$ and is defined as

$$(3) \qquad \mathcal{E} = \{x \in \mathbb{R}^n : (x - c)^T Q (x - c) \leq 1\}.$$

The matrix Q determines the shape and the orientation of $\mathcal{E}$. In particular, the axes of $\mathcal{E}$ are the eigenvectors $d^1, \ldots, d^n \in \mathbb{R}^n$ of Q, and the length of each axis is given by $1/\sqrt{\lambda_1}, \ldots, 1/\sqrt{\lambda_n}$, where $\lambda_1, \ldots, \lambda_n$ are the corresponding eigenvalues of Q. Therefore, the volume of $\mathcal{E}$, denoted by vol $\mathcal{E}$, is given by

$$(4) \qquad \text{vol } \mathcal{E} = \eta \, \det Q^{-\frac{1}{2}} = \eta \left(1/\sqrt{\Pi_{i=1}^n \lambda_i} \right),$$

where η is the volume of the unit ball in $\mathbb{R}^n$ [12]. Note that an ellipsoid $\mathcal{E}$ induces a norm on $\mathbb{R}^n$ via $\|x\|_{\mathcal{E}} := (x^T Q x)^{1/2}$. Therefore, every ellipsoid can be viewed as a translation of the unit ball in terms of the ellipsoidal norm induced by it.

Throughout this paper, we will assume that each of the input ellipsoids $\mathcal{E}_1, \ldots, \mathcal{E}_m \subset \mathbb{R}^n$ is full-dimensional. Note that this assumption is without loss of generality since any lower-dimensional ellipsoid can easily be approximated by a "thin" full-dimensional one. We remark that this assumption is merely for technical convenience, which allows us to have a uniform representation of each of the ellipsoids in the form given by (3). In addition, this assumption guarantees that $\text{conv}(\cup_{i=1}^m \mathcal{E}_i)$ is full-dimensional and leads to a simpler characterization of the ellipsoid containment problem (see Proposition 2.7). In particular, the full-dimensionality assumption on each of the ellipsoids can be relaxed by the weaker assumption that $\text{conv}(\cup_{i=1}^m \mathcal{E}_i)$ is full-dimensional and our analysis would still carry over to this slightly more general setting (see the discussion after Proposition 2.6). We refer the reader to [2] for further discussions on extremal ellipsoids.

We start with a classical result on the quality of the approximation of $\text{MVCE}(\mathcal{S})$ of a convex set $\mathcal{S} \subset \mathbb{R}^n$.

THEOREM 2.1 (Löwner–John [13]). *Let $\mathcal{S} \subset \mathbb{R}^n$ be a compact, convex set with a nonempty interior. Then, $MVCE(\mathcal{S})$ exists and is unique and satisfies*

$$(5) \qquad \frac{1}{n} MVCE(\mathcal{S}) \subseteq \mathcal{S} \subseteq MVCE(\mathcal{S}),$$

where the ellipsoid on the left-hand side is obtained by scaling $MVCE(\mathcal{S})$ around its center by a factor of $1/n$. Furthermore, if $\mathcal{S}$ is symmetric around the origin, then the factor on the left-hand side of (5) can be improved to $1/\sqrt{n}$.

We next state a well-known lemma that will be useful for our analysis.

LEMMA 2.2 (Schur complement). *Let*

$$A = \begin{bmatrix} B & C \\ C^T & D \end{bmatrix}$$

be a symmetric matrix with $B \in \mathcal{S}^\alpha$ and $D \in \mathcal{S}^\beta$. Assume that $D \succ 0$. Then, $A \succeq 0$ if and only if $B - CD^{-1}C^T \succeq 0$.

2.1. Complexity of semidefinite feasibility and optimization. Consider the following feasibility problems:

1. **(PF)** Given $A_1, A_2, \ldots, A_\kappa \in \mathcal{S}^n$ and $\beta_1, \ldots, \beta_\kappa \in \mathbb{R}$, determine whether there exists a matrix $X \in \mathcal{S}^n$ such that

$$A_i \bullet X \le \beta_i, \quad i = 1, \ldots, \kappa, \quad X \succeq 0.$$

2. **(DF)** Given $B_0, B_1, \ldots, B_\kappa \in \mathcal{S}^n$, determine whether there exist real numbers $y_1, \ldots, y_\kappa$ such that

$$B_0 + y_1 B_1 + y_2 B_2 + \cdots + y_\kappa B_\kappa \succeq 0.$$

The complexity of the problems **(PF)** and **(DF)** is still a fundamental open problem. In the real number model of computation, both problems are in NP since one can check in polynomial time whether a given symmetric matrix is positive semidefinite using Cholesky factorization. Ramana [22] proved that both problems belong to NP $\cap$ co-NP. Porkolab and Khachiyan [21] established the following complexity results, which, in turn, are mainly based on the first-order theory of the reals developed by Renegar [24].

THEOREM 2.3. *Problems* **(PF)** *and* **(DF)** *can be solved in* $\kappa n^{O(\min\{\kappa, n^2\})}$ *and* $O(\kappa n^4) + n^{O(\min\{\kappa, n^2\})}$ *operations over the reals, respectively.*

In addition, let us consider the following optimization versions:

1. **(PO)** Given $D, A_1, A_2, \ldots, A_\kappa \in \mathcal{S}^n$ and $\beta_1, \ldots, \beta_\kappa \in \mathbb{R}$, solve

$$\alpha^* := \inf_{X \in \mathcal{S}^n} \ \{D \bullet X : A_i \bullet X \le \beta_i, \quad i = 1, \ldots, \kappa, \quad X \succeq 0\}.$$

2. **(DO)** Given $B_0, B_1, \ldots, B_\kappa \in \mathcal{S}^n$ and $d \in \mathbb{R}^\kappa$, solve

$$\beta^* := \sup_{y_1, \ldots, y_\kappa \in \mathbb{R}} \ \left\{ \sum_{i=1}^\kappa d_i y_i : B_0 + y_1 B_1 + y_2 B_2 + \cdots + y_\kappa B_\kappa \succeq 0 \right\}.$$

The complexity results of Theorem 2.3 also extend to the optimization versions **(PO)** and **(DO)** [21].

THEOREM 2.4. *For problems* **(PO)** *and* **(DO)***, each of the following can be solved in* $\kappa n^{O(\min\{\kappa, n^2\})}$ *and* $O(\kappa n^4) + n^{O(\min\{\kappa, n^2\})}$ *operations over the reals, respectively:* (i) *feasibility,* (ii) *boundedness,* (iii) *attainment of the optimal value, and* (iv) *computation of a least norm optimal solution.*

One important consequence of Theorems 2.3 and 2.4 is that semidefinite feasibility and semidefinite optimization can be solved in polynomial time if κ is fixed. We state this as a separate corollary.

COROLLARY 2.5. *Each of the four problems* **(PF)**, **(DF)**, **(PO)**, *and* **(DO)** *can be solved in polynomial time for fixed* κ.

This result will play a key role in our algorithm as the semidefinite feasibility and semidefinite optimization problems we will encounter will always satisfy the condition of the corollary.

2.2. Ellipsoid containment. In this section, we study the problem of deciding whether a given full-dimensional ellipsoid $\mathcal{E}$ is contained in another full-dimensional ellipsoid $\mathcal{E}^*$. Furthermore, we establish how to efficiently compute a point in $\mathcal{E}$ that is furthest from the center of $\mathcal{E}^*$ in terms of the ellipsoidal norm induced by $\mathcal{E}^*$.

We start with the following well-known result about polynomiality of quadratic optimization over an ellipsoid (see, e.g., [34]). We remark that this result can be found elsewhere in the literature (see, e.g., [27, 23, 28, 36, 6]). We mainly include it here for the sake of completeness. Our treatment can be considered as a special case of the more general proof of [28] and relies on the fact that the possibly nonconvex optimization problem admits a tight semidefinite programming (SDP) relaxation, whose optimal solution can be used to recover an optimal solution for the original problem.

PROPOSITION 2.6. *Any quadratic function $f : \mathbb{R}^n \to \mathbb{R}$ can be maximized over a full-dimensional ellipsoid in $O(n^{O(1)})$ operations, where $O(1)$ is a universal constant greater than three.*

Proof. Let $f(x) := x^T A x + 2b^T x + \gamma$, where $A \in \mathcal{S}^n$, $b \in \mathbb{R}^n$, and $\gamma \in \mathbb{R}$, and let $\mathcal{E} \subset \mathbb{R}^n$ denote a full-dimensional ellipsoid, which admits a representation given by $\mathcal{E} := \{x \in \mathbb{R}^n : (x - c)^T Q(x - c) \leq 1\}$, where $Q \in \mathcal{S}^n$ is positive definite and $c \in \mathbb{R}^n$. We wish to solve

$$\text{(P)} \quad \max_{x \in \mathbb{R}^n} \ \{f(x) : x \in \mathcal{E}\}.$$

We consider the following SDP relaxation:

$$\text{(SP)} \quad \max_{X \in \mathcal{S}^{n+1}} \ \{F \bullet X : G \bullet X \leq 0, \ E_{n+1} \bullet X = 1, \ X \succeq 0\},$$

where

$$F := \begin{bmatrix} A & b \\ b^T & \gamma \end{bmatrix}, \quad G := \begin{bmatrix} Q & -Qc \\ -c^T Q & c^T Q c - 1 \end{bmatrix}, \quad E_{n+1} = e^{n+1}(e^{n+1})^T.$$

Note that (SP) is a relaxation of (P) since for any feasible solution $x \in \mathbb{R}^n$ of (P),

$$\begin{bmatrix} x \\ 1 \end{bmatrix} [x^T \ \ 1] = \begin{bmatrix} xx^T & x \\ x^T & 1 \end{bmatrix} \succeq 0$$

is a feasible solution of (SP) with the same objective function value. We claim that the relaxation is exact in the sense that the optimal values of (P) and (SP) coincide and an optimal solution of (SP) can be converted into an optimal solution of (P).

Consider the following Lagrangian dual of (SP):

$$\text{(SD)} \quad \min_{\lambda, \beta} \{\beta : \ \lambda G + \beta E_{n+1} \succeq F, \ \lambda \geq 0\}.$$

We now make several observations about (SP) and (SD). Note that (SP) satisfies the Slater condition since the solution given by

$$\tilde{X} := \begin{bmatrix} cc^T + \alpha I & c \\ c^T & 1 \end{bmatrix}$$

satisfies $E_{n+1} \bullet \tilde{X} = 1$, $G \bullet \tilde{X} = -1 + \alpha Q \bullet I < 0$, for sufficiently small $\alpha > 0$, and $\tilde{X} \succ 0$, which implies that $\tilde{X}$ is a strictly feasible solution of (SP). Therefore, strong duality holds between (SP) and (SD), and the optimal value is attained in (SD). Furthermore, the feasible set of (SP) is bounded because the only solution to the system

$$G \bullet Y \leq 0, \quad E_{n+1} \bullet Y = 0, \quad Y \succeq 0, \quad Y \in \mathcal{S}^{n+1}$$

is $Y = 0$ since $Q \succ 0$. Therefore, the optimal value is also attained in (SP).

By Corollary 2.5, we can solve (SP) in $O(n^{O(1)})$ time (one can replace the equality constraint with two inequality constraints). Let X^* and (λ^*, β^*) denote optimal solutions of (SP) and (SD), respectively. It follows from optimality conditions that

$$(6) \qquad X^* \bullet (\lambda^* G + \beta^* E_{n+1} - F) = 0, \quad \lambda^*(G \bullet X^*) = 0.$$

Since $G \bullet X^* \leq 0$, we can compute a rank-one decomposition of $X^* := \sum_{i=1}^{\rho} p^i (p^i)^T$, where $\rho := \mathrm{rank}(X^*) \geq 1$ and $p^i \in \mathbb{R}^{n+1}$, $p^i \neq 0$, $i = 1, \ldots, \rho$, in $O(n^3)$ operations such that $(p^i)^T G p^i \leq 0$, $i = 1, \ldots, \rho$ [28, Proposition 3]. We now construct a rank-one optimal solution of (SP) using this decomposition.

By (6), $\sum_{i=1}^{\rho} (p^i)^T (\lambda^* G + \beta^* E_{n+1} - F) p^i = 0$, which implies that

$$(7) \qquad (p^i)^T (\lambda^* G + \beta^* E_{n+1} - F) p^i = 0, \quad i = 1, \ldots, \rho,$$

by dual feasibility. Similarly, $\lambda^*(G \bullet X^*) = \lambda^* \sum_{i=1}^{\rho} (p^i)^T G p^i = 0$, which implies that

$$(8) \qquad \lambda^*(p^i)^T G p^i = 0, \quad i = 1, \ldots, \rho,$$

since $(p^i)^T G p^i \leq 0$, $i = 1, \ldots, \rho$, and $\lambda^* \geq 0$.

Let j be any index in $\{1, 2, \ldots, \rho\}$ and let us define

$$p^j = \left[\begin{array}{c} x^j \\ \alpha^j \end{array} \right],$$

where $x^j \in \mathbb{R}^n$ and $\alpha^j \in \mathbb{R}$. We claim that $\alpha^j \neq 0$. Otherwise, $0 \geq (p^j)^T G p^j = (x^j)^T Q x^j$, which implies that $x^j = 0$ since $Q \succ 0$, contradicting the fact that $p^j \neq 0$. We now let $x_*^j := (1/\alpha^j) p^j$. Since $G \bullet x_*^j (x_*^j)^T \leq 0$ and $E_{n+1} \bullet x_*^j (x_*^j)^T = 1$, it follows from (7) and (8) that $x_*^j (x_*^j)^T$ is a rank-one optimal solution of (SP), which implies that $(1/\alpha^j) x^j$ is an optimal solution of (P). (We remark that each of the indices in $\{1, 2, \ldots, \rho\}$ can be used to compute a different optimal solution of (P).) $\qquad \square$

In fact, Proposition 2.6 holds true even if the ellipsoid defining the feasible region of the optimization problem is lower-dimensional. In this case, one can restrict the quadratic function f to the smallest affine subspace containing the ellipsoid and invoke the same analysis in the proof. We now use Proposition 2.6 to give a simple proof of the well-known characterization of the ellipsoid containment problem.

PROPOSITION 2.7. *Let $\mathcal{E} \subset \mathbb{R}^n$ and $\mathcal{E}^* \subset \mathbb{R}^n$ denote two full-dimensional ellipsoids with representations given by $\mathcal{E} := \{x \in \mathbb{R}^n : (x - c)^T Q (x - c) \leq 1\}$ and $\mathcal{E}^* := \{x \in \mathbb{R}^n : (x - c^*)^T Q^* (x - c^*) \leq 1\}$, where $Q \in \mathcal{S}^n$ and $Q^* \in \mathcal{S}^n$ are positive definite and $c \in \mathbb{R}^n$ and $c^* \in \mathbb{R}^n$. Then, $\mathcal{E} \subseteq \mathcal{E}^*$ if and only if there exists $\tau > 0$ such that*

$$(9) \qquad \tau \left[\begin{array}{cc} Q & -Qc \\ -c^T Q & c^T Q c - 1 \end{array} \right] \succeq \left[\begin{array}{cc} Q^* & -Q^* c^* \\ -c^{*T} Q^* & c^{*T} Q^* c^* - 1 \end{array} \right].$$

Proof. The statement follows directly from the $\mathcal{S}$-lemma [33] (see also [20] for a comprehensive treatment). However, we give a simple proof using standard duality arguments.

If (9) is satisfied, then we must have $\tau > 0$ since $Q \succ 0$ and $Q^* \succ 0$. Consider

$$(P) \quad \max_{x \in \mathbb{R}^n} \{(x - c^*)^T Q^* (x - c^*) - 1 : (x - c)^T Q (x - c) - 1 \leq 0\}.$$

By an argument similar to that in the proof of Proposition 2.6, it follows that

$$\text{(SP)} \quad \max_{X \in \mathcal{S}^{n+1}} \{F \bullet X : \; G \bullet X \leq 0, \; E_{n+1} \bullet X = 1, \; X \succeq 0\}$$

is a tight SDP relaxation of (P), where $F \in \mathcal{S}^{n+1}$ and $G \in \mathcal{S}^{n+1}$ are respectively given by

$$F = \begin{bmatrix} Q^* & -Q^*c^* \\ -c^{*T}Q^* & c^{*T}Q^*c^* - 1 \end{bmatrix}, \quad G = \begin{bmatrix} Q & -Qc \\ -c^T Q & c^T Qc - 1 \end{bmatrix}.$$

The dual of (SP) is

$$\text{(SD)} \quad \min_{\lambda, \beta} \{\beta : \; \lambda G + \beta E_{n+1} \succeq F, \; \lambda \geq 0\}.$$

Let $v(P), v(SP)$, and $v(SD)$ denote the optimal values of (P), (SP), and (SD), respectively. It follows from the proof of Proposition 2.6 that

$$(10) \qquad\qquad\qquad v(P) = v(SP) = v(SD).$$

Obviously, $\mathcal{E} \subseteq \mathcal{E}^*$ if and only if $v(P) \leq 0$. If (9) is feasible, then $(\lambda, \beta) = (\tau, 0)$ is a feasible solution of (SD), which implies that $v(P) = v(SD) \leq 0$. Conversely, if $v(P) \leq 0$, then let (λ^*, β^*) be an optimal solution of (SD) with optimal value $v(SD) = v(P) = \beta^* \leq 0$. Then

$$\lambda^* G \succeq \lambda^* G + \beta^* E_{n+1} \succeq F,$$

since $E_{n+1} \succeq 0$ and $\beta^* \leq 0$, which implies that λ^* is a feasible solution of (9). This completes the proof. $\quad\square$

We close this subsection by giving an equivalent characterization of (9).

LEMMA 2.8. *Condition* (9) *is equivalent to*

$$(11) \qquad \tau \begin{bmatrix} Q & -Qc & 0 \\ -c^T Q & c^T Qc - 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \succeq \begin{bmatrix} Q^* & -Q^*c^* & 0 \\ -c^{*T}Q^* & -1 & c^{*T}Q^* \\ 0 & Q^*c^* & -Q^* \end{bmatrix}.$$

Proof. We use the notation of Lemma 2.2. After rewriting (11) as a constraint of the form $A \succeq 0$, we let B denote the top left 2×2 block and define C and D accordingly. The equivalence now simply follows from the Schur complement lemma since $D := Q^* \succ 0$. $\quad\square$

We remark that condition (9) (or, equivalently, condition (11)) is a semidefinite constraint in a single variable. Therefore, it follows from Corollary 2.5 that ellipsoid containment can be checked in polynomial time.

It follows from (11) that the problem of computing the MVCE of a set of m full-dimensional ellipsoids can be formulated as a convex determinant maximization problem (see, e.g., [5, 6, 2]) with m linear matrix inequalities of size $(2n+1) \times (2n+1)$, m nonnegative variables $\tau_1, \ldots, \tau_m$, an $n \times n$ positive definite matrix variable Q^* that determines the shape and the orientation of the optimal ellipsoid, and an n-dimensional vector variable $z^* := Q^*c^*$, from which the center of the optimal ellipsoid can be recovered. As the dimension of the problem grows, the computational cost of interior-point algorithms [32, 31] quickly becomes prohibitive. This is one of our motivations to develop a specialized algorithm for the MVCE problem.

3. Initial volume approximation. Let $\mathcal{E}_1, \ldots, \mathcal{E}_m$ denote m full-dimensional ellipsoids, which admit representations given by

$$(12) \qquad \mathcal{E}_i := \{x \in \mathbb{R}^n : (x - c^i)^T Q^i (x - c^i) \leq 1\}, \; i = 1, \ldots, m,$$

where $Q^i \in \mathcal{S}^n$ is positive definite and $c^i \in \mathbb{R}^n$, $i = 1, \ldots, m$. We define $\mathcal{S} := \text{conv}\,(\cup_{i=1}^m \mathcal{E}_i)$. In this section, we present a simple deterministic algorithm that identifies a finite subset $\mathcal{X}_0 \subset \cup_{i=1}^m \mathcal{E}_i$ of size $2n$ such that vol $\text{MVCE}(\mathcal{X}_0)$ is a provable approximation to vol $\text{MVCE}(\mathcal{S})$.

ALGORITHM 3.1 (volume approximation algorithm).
Require: Input set $\mathcal{E}_1, \ldots, \mathcal{E}_m \subset \mathbb{R}^n$
1: $\Psi \leftarrow \{0\}$, $\mathcal{X}_0 \leftarrow \emptyset$, $k \leftarrow 0$.
2: While $\mathbb{R}^n \setminus \Psi \neq \emptyset$ do
3: **loop**
4: $\quad k \leftarrow k+1$. Pick an arbitrary unit vector $b^k \in \mathbb{R}^n$ in the orthogonal complement of Ψ.
5: $\quad x^{2k-1} \leftarrow \arg\max_{i=1,\ldots,m}\{(b^k)^T x : x \in \mathcal{E}_i\}$, $\mathcal{X}_0 \leftarrow \mathcal{X}_0 \cup \{x^{2k-1}\}$.
6: $\quad x^{2k} \leftarrow \arg\min_{i=1,\ldots,m}\{(b^k)^T x : x \in \mathcal{E}_i\}$, $\mathcal{X}_0 \leftarrow \mathcal{X}_0 \cup \{x^{2k}\}$.
7: $\quad \Psi \leftarrow \text{span}(\Psi, \{x^{2k-1} - x^{2k}\})$.
8: **end loop**
9: **Output:** $\mathcal{X}_0$

LEMMA 3.1. *Algorithm* 3.1 *terminates after* $O(mn^3)$ *arithmetic operations and returns a subset* $\mathcal{X}_0 \subset \cup_{i=1}^m \mathcal{E}_i$ *with* $|\mathcal{X}_0| = 2n$ *such that*

$$(13) \qquad \text{vol MVCE}(\mathcal{S}) \leq n^{2n}\text{vol MVCE}(\mathcal{X}_0).$$

Proof. We first establish the running time of Algorithm 3.1. At step k, Ψ is given by the span of k linearly independent vectors since $\mathcal{S}$ is full-dimensional. Hence, upon termination, $\Psi = \mathbb{R}^n$. It follows that $|\mathcal{X}_0| = 2n$. At each step, we optimize a linear function over each of the m ellipsoids $\mathcal{E}_i$. Let $Q^i = (U^i)^T U^i$, $i = 1, \ldots, m$, denote the Cholesky factorization of Q^i, $i = 1, \ldots, m$, which can be computed in $O(mn^3)$ operations. Note that $\mathcal{E}_i = \{x \in \mathbb{R}^n : x = (U^i)^{-1} u + c^i, \|u\| \leq 1\}$, $i = 1, \ldots, m$. Therefore, at step k, each optimization problem has a closed form solution given by $\tilde{x}^{i,k}_{\max,\min} := c^i \pm (1/\|(U^i)^{-T} b^k\|)(U^i)^{-1}(U^i)^{-T} b^k$ with an optimal value of $(b^k)^T c^i \pm (1/\|(U^i)^{-T} b^k\|)(b^k)^T (U^i)^{-1}(U^i)^{-T} b^k$. For each ellipsoid $\mathcal{E}_i$, $\tilde{x}^{i,k}_{\max,\min}$ can be computed in $O(n^2)$ operations since U^i is upper triangular, which yields an overall computational cost of $O(mn^3)$ operations after n steps. Therefore, Algorithm 3.1 terminates after $O(mn^3)$ arithmetic operations.

We now prove (13). It follows from the results of Betke and Henk [3] that vol $\mathcal{S} \leq n!$ vol conv$(\mathcal{X}_0)$. Combining this inequality with Theorem 2.1, we obtain

$$\frac{1}{n^n} \text{ vol MVCE}(\mathcal{S}) \leq \text{vol } \mathcal{S} \leq n! \text{ vol conv}(\mathcal{X}_0) \leq n! \text{ vol MVCE}(\mathcal{X}_0),$$

which implies that vol $\text{MVCE}(\mathcal{S}) \leq n!n^n\text{vol MVCE}(\mathcal{X}_0) \leq n^{2n}\text{vol MVCE}(\mathcal{X}_0)$. $\quad\square$

4. A first-order algorithm. In this section, we present a first-order algorithm to compute a $(1 + \epsilon)$-approximation to the MVCE of the union of a set of full-dimensional ellipsoids $\mathcal{E}_1, \ldots, \mathcal{E}_m \subset \mathbb{R}^n$ for $\epsilon > 0$. Our algorithm is a generalization of the first-order algorithm presented in [18] to compute the MVCE of a finite set of m points, which, in turn, is obtained from a modification of Khachiyan's algorithm [15].

Our treatment closely follows the interpretation of Khachiyan's algorithm presented in [18].

As a by-product, we establish the existence of a finite core set $\mathcal{X} \subset \cup_{i=1,\ldots,m} \mathcal{E}_i$ whose size depends on only the dimension n and the parameter ϵ, but is independent of the number of ellipsoids m.

ALGORITHM 4.1 (a first-order algorithm that computes a $(1+\epsilon)$-approximation to MVCE($\mathcal{S}$)).

Require: Input set of ellipsoids $\mathcal{E}_1, \ldots, \mathcal{E}_m \subset \mathbb{R}^n$ given by (12) and $\epsilon > 0$.
1: Run Algorithm 3.1 on $\mathcal{E}_1, \ldots, \mathcal{E}_m$ to obtain output $\mathcal{X}_0 := \{x^1, \ldots, x^{2n}\}$.
2: $u^0 \leftarrow (1/2n)e \in \mathbb{R}^{2n}$.
3: $w^0 \leftarrow \sum_{j=1}^{2n} x^j u_j^0$.
4: $(M^0)^{-1} \leftarrow n \sum_{j=1}^{2n} u_j^0 (x^j - w^0)(x^j - w^0)^T$.
5: $\mathcal{F}_0 \leftarrow \{x \in \mathbb{R}^n : (x - w^0)^T M^0 (x - w^0) \leq 1\}$.
6: $x^{2n+1} \leftarrow \arg\max_{i=1,\ldots,m}\{(x - w^0)^T M^0 (x - w^0) : x \in \mathcal{E}_i\}$.
7: $\epsilon_0 \leftarrow (x^{2n+1} - w^0)^T M^0 (x^{2n+1} - w^0) - 1$.
8: $k \leftarrow 0$.
9: While $\epsilon_k > (1+\epsilon)^{2/n} - 1$ do
10: **loop**
11: $\quad \beta_k \leftarrow \frac{\epsilon_k}{(n+1)(1+\epsilon_k)}$.
12: $\quad k \leftarrow k + 1$.
13: $\quad u^k \leftarrow \begin{bmatrix} (1 - \beta_{k-1})u^{k-1} \\ \beta_{k-1} \end{bmatrix}$.
14: $\quad w^k \leftarrow \sum_{j=1}^{2n+k} x^j u_j^k$.
15: $\quad (M^k)^{-1} \leftarrow n \sum_{j=1}^{2n+k} u_j^k (x^j - w^k)(x^j - w^k)^T$.
16: $\quad \mathcal{F}_k \leftarrow \{x \in \mathbb{R}^n : (x - w^k)^T M^k (x - w^k) \leq 1\}$.
17: $\quad \mathcal{X}_k \leftarrow \mathcal{X}_{k-1} \cup \{x^{2n+k}\}$.
18: $\quad x^{2n+k+1} \leftarrow \arg\max_{i=1,\ldots,m}\{(x - w^k)^T M^k (x - w^k) : x \in \mathcal{E}_i\}$.
19: $\quad \epsilon_k \leftarrow (x^{2n+k+1} - w^k)^T M^k (x^{2n+k+1} - w^k) - 1$.
20: **end loop**
21: **Output:** $\sqrt{1 + \epsilon_k}\,\mathcal{F}_k$, $\mathcal{X}_k$

We now describe Algorithm 4.1. Given m full-dimensional ellipsoids $\mathcal{E}_1, \ldots, \mathcal{E}_m \subset \mathbb{R}^n$ with representations given by (12), the algorithm calls Algorithm 3.1 and computes a finite set $\mathcal{X}_0 \subset \cup_{i=1}^m \mathcal{E}_i$ with $|\mathcal{X}_0| = 2n$. Next, a "trial ellipsoid" $\mathcal{F}_0$ is defined. Note that the center w^0 of $\mathcal{F}_0$ is simply the sample mean of $\mathcal{X}_0$ and M^0 is the inverse of the (scaled) sample covariance matrix of $\mathcal{X}_0$. ϵ_k measures the extent to which $\mathcal{F}_k$ should be enlarged around its center in order to cover $\mathcal{S} := \text{conv}(\cup_{i=1}^m \mathcal{E}_i)$. u^k can be viewed as a nonnegative weight vector whose components sum up to one. Note that the dimension of u^k increases by one at each iteration and is equal to $|\mathcal{X}_k|$. Unless the termination criterion is satisfied, the algorithm proceeds in an iterative manner as follows: At Step 13, u^k gets updated and is used to define w^k and M^k for the next trial ellipsoid $\mathcal{F}_k$. Observe that x^{2n+k} is precisely the farthest point in $\mathcal{S}$ from the center of the trial ellipsoid $\mathcal{F}_{k-1}$ in terms of its ellipsoidal norm. It is straightforward to verify that

$$(14) \qquad w^k = (1 - \beta_{k-1})w^{k-1} + \beta_{k-1}x^{2n+k}, \quad k = 1, 2, \ldots,$$

and

$$(15) \qquad (M^k)^{-1} = (1 - \beta_{k-1})(M^{k-1})^{-1} + n(1 - \beta_{k-1})\beta_{k-1}d^k(d^k)^T$$

for $k = 1, 2, \ldots$, where $d^k := x^{2n+k} - w^{k-1}$. It follows that the next trial ellipsoid $\mathcal{F}_k$ is obtained by shifting the center of $\mathcal{F}_{k-1}$ towards x^{2n+k}, and its shape is determined by a nonnegative combination of $(M^{k-1})^{-1}$ and a rank-one update. This update can be viewed as "enriching the eigenspace of $(M^{k-1})^{-1}$ in the direction $d^k := x^{2n+k} - w^{k-1}$." We refer the reader to [18, section 4.3] for a related discussion. The parameter $\beta_{k-1} \in [0, 1)$ solves the following line search problem as observed by Khachiyan [15]:

$$(\text{LS}(k)) \quad \max_{\beta \in [0,1]} \log \det \left[(1 - \beta)(M^{k-1})^{-1} + n(1 - \beta)\beta d^k (d^k)^T \right]$$

for $k = 1, 2, \ldots$, where $d^k := x^{2n+k} - w^{k-1}$. Algorithm 4.1 terminates when the desired accuracy is achieved.

Algorithm 4.1 is an extension of the one proposed in [18] that computes a $(1 + \epsilon)$-approximation to the MVCE of a finite set of m *points* in $\mathbb{R}^n$, which, in turn, is a modification of Khachiyan's algorithm [15]. The algorithm in [15] can be viewed as a sequential linear programming algorithm (or, equivalently, as a Frank–Wolfe algorithm [29]) for the nonlinear optimization problem arising from the dual formulation of the MVCE problem (cf. $(\mathbf{D}(\mathcal{X_0}))$ in the proof of Theorem 4.1) for a finite set of points (see, e.g., the discussion in [18, section 4.1]). Algorithm 4.1 is motivated by the simple observation that the union of a set of ellipsoids in $\mathbb{R}^n$ can be viewed as an infinite set of points. Despite the fact that the finite-dimensional optimization formulation on which Khachiyan's algorithm is based no longer carries over to this more general setting, our main goal in this paper is to establish that essentially the same framework can be used with proper modifications to approximate the MVCE of the union of a finite number of ellipsoids. Since the algorithm is driven by linearizing the nonlinear objective function of the dual optimization formulation, we continue to refer to Algorithm 4.1 as a first-order algorithm. We remark that the interior-point algorithms of [32, 31] also rely on the second-order information arising from the Hessian of the objective function.

We next analyze the complexity of Algorithm 4.1. Our analysis resembles those of Khachiyan [15] and Kumar and Yıldırım [18]. The key ingredient in the complexity analysis is to demonstrate that Algorithm 4.1 produces a sequence $\{\mathcal{F}_k\}$ of trial ellipsoids with strictly increasing volumes. We utilize Lemma 3.1 to show that $\operatorname{vol} \mathcal{F}_0$ is already a provable approximation to $\operatorname{vol} \operatorname{MVCE}(\mathcal{S})$. The analysis will then be complete by establishing that each step of Algorithm 4.1 can be executed efficiently.

We start by proving that $\operatorname{vol} \mathcal{F}_0$ is a provable approximation to $\operatorname{vol} \operatorname{MVCE}(\mathcal{S})$.

THEOREM 4.1. *The ellipsoid $\mathcal{F}_0 \subset \mathbb{R}^n$ defined in Algorithm* 4.1 *satisfies*

$$(16) \qquad \log \operatorname{vol} \mathcal{F}_0 \leq \log \operatorname{vol} \operatorname{MVCE}(\mathcal{S}) \leq \log \operatorname{vol} \mathcal{F}_0 + 2n \log n + \frac{n}{2} \log 2.$$

Proof. We first establish that

$$(17) \qquad \log \operatorname{vol} \mathcal{F}_0 \leq \log \operatorname{vol} \operatorname{MVCE}(\mathcal{X}_0),$$

where $\mathcal{X}_0 = \{x^1, \ldots, x^{2n}\}$ denotes the set of $2n$ points returned by Algorithm 3.1. Consider the following dual formulation to compute $\operatorname{MVCE}(\mathcal{X}_0)$ (see, e.g., [15] or [29]):

$$
\begin{aligned}
(\mathbf{D}(\mathcal{X_0})) \quad \max_u \quad & \log \det \Pi_0(u) \\
\text{s.t.} \quad & e^T u = 1, \\
& u \geq 0,
\end{aligned}
$$

where $u \in \mathbb{R}^{2n}$ is the decision variable and $\Pi_0 : \mathbb{R}^{2n} \to \mathcal{S}^{n+1}$ is a linear operator given by

$$\Pi_0(u) := \sum_{j=1}^{2n} u_j \left[\begin{array}{cc} x^j(x^j)^T & x^j \\ (x^j)^T & 1 \end{array} \right].$$

$\mathrm{MVCE}(\mathcal{X}_0)$ can be recovered from an optimal solution u^* of $(\mathbf{D}(\mathcal{X}_0))$ [18, Lemma 2.1]. Furthermore, the optimal value of $(\mathbf{D}(\mathcal{X}_0))$ satisfies

$$(19) \qquad \log \mathrm{vol}\, \mathrm{MVCE}(\mathcal{X}_0) = \log \eta + \frac{n}{2} \log n + \frac{1}{2} \log \det \Pi_0(u^*),$$

where η is the volume of the unit ball in $\mathbb{R}^n$.

Let us consider the feasible solution $u^0 := (1/2n)e \in \mathbb{R}^{2n}$ of $(\mathbf{D}(\mathcal{X}_0))$. We have

$$(20) \qquad \begin{aligned} \Pi_0(u^0) &= \left[\begin{array}{cc} (1/2n)\sum_{j=1}^{2n} x^j(x^j)^T & w^0 \\ (w^0)^T & 1 \end{array} \right], \\ &= \left[\begin{array}{cc} I & w^0 \\ 0 & 1 \end{array} \right] \left[\begin{array}{cc} (1/n)(M^0)^{-1} & 0 \\ 0 & 1 \end{array} \right] \left[\begin{array}{cc} I & 0 \\ (w^0)^T & 1 \end{array} \right], \end{aligned}$$

which implies that

$$(21) \quad \log \det \Pi_0(u^0) = -n \log n + \log \det(M^0)^{-1} = -n \log n + 2 \log \det(M^0)^{-1/2}.$$

However, $\log \mathrm{vol}\, \mathcal{F}_0 = \log \eta + \log \det(M^0)^{-1/2}$. Combining this equality with (21), we obtain

$$\log \mathrm{vol}\, \mathcal{F}_0 = \log \eta + \frac{n}{2} \log n + \frac{1}{2} \log \det \Pi_0(u^0).$$

Since u^0 is a feasible solution for the maximization problem $(\mathbf{D}(\mathcal{X}_0))$, it follows from (19) that $\log \mathrm{vol}\, \mathcal{F}_0 \leq \log \mathrm{vol}\, \mathrm{MVCE}(\mathcal{X}_0)$.

Since $\mathcal{X}_0 \subset \mathcal{S}$, we clearly have $\log \mathrm{vol}\, \mathrm{MVCE}(\mathcal{X}_0) \leq \log \mathrm{vol}\, \mathrm{MVCE}(\mathcal{S})$, which proves the first inequality in (16). To prove the second inequality, let

$$B := [x^1, \ldots, x^{2n}] \in \mathbb{R}^{n \times 2n}.$$

Then, $w^0 = (1/2n)Be$ and it is easy to verify that $(M^0)^{-1} = (1/2)BPB^T$, where $P := I - (1/2n)ee^T$ is an orthogonal projection matrix onto the orthogonal complement of the vector e. Note that $Pe^j = e^j - (1/2n)e$, $j = 1, \ldots, 2n$. Therefore, for any $j = 1, \ldots, 2n$, we have

$$\begin{aligned} (x^j - w^0)^T M^0 (x^j - w^0) &= 2(e^j - (1/2n)e)^T B^T (BPB^T)^{-1} B(e^j - (1/2n)e), \\ &= 2(Pe^j)^T PB^T (BP^2B^T)^{-1} BP(Pe^j), \\ &\leq 2\|Pe^j\|^2, \\ &= \frac{2n-1}{n}, \\ &< 2, \end{aligned}$$

where we used $P = P^2$ on the second line and the fact that $PB^T(BP^2B^T)^{-1}BP$ is an orthogonal projection matrix to derive the first inequality. Consequently, the

ellipsoid $\mathcal{G} := \{x \in \mathbb{R}^n : (x - w^0)^T (1/2) M^0 (x - w^0) \leq 1\}$ covers $\mathcal{X}_0$. Therefore,

$$\log \mathrm{vol}\, \mathrm{MVCE}(\mathcal{X}_0) \leq \log \mathrm{vol}\, \mathcal{G},$$
$$= \log \eta + \frac{n}{2} \log 2 + \log \det(M^0)^{-1/2},$$
$$= \frac{n}{2} \log 2 + \log \mathrm{vol}\, \mathcal{F}_0,$$

which implies that $\log \mathrm{vol}\, \mathcal{F}_0 \geq \log \mathrm{vol}\, \mathrm{MVCE}(\mathcal{X}_0) - (n/2) \log 2$. By Lemma 3.1, we have $\log \mathrm{vol}\, \mathrm{MVCE}(\mathcal{X}_0) \geq \log \mathrm{vol}\, \mathrm{MVCE}(\mathcal{S}) - 2n \log n$. Combining these two inequalities, we obtain $\log \mathrm{vol}\, \mathcal{F}_0 + 2n \log n + (n/2) \log 2 \geq \log \mathrm{vol}\, \mathrm{MVCE}(\mathcal{S})$ as desired. $\qquad \square$

The next lemma relates $\log \mathrm{vol}\, \mathcal{F}_k$ to $\log \mathrm{vol}\, \mathrm{MVCE}(\mathcal{S})$.

LEMMA 4.2. *For any $k = 0, 1, 2, \ldots$, we have*

$$(22) \qquad \log \mathrm{vol}\, \mathcal{F}_k \leq \log \mathrm{vol}\, \mathrm{MVCE}(\mathcal{S}) \leq \log \mathrm{vol}\, \mathcal{F}_k + \frac{n}{2} \log(1 + \epsilon_k).$$

Proof. By definition of ϵ_k, $\sqrt{1 + \epsilon_k}\, \mathcal{F}_k \supseteq \mathcal{S}$, where $\sqrt{1 + \epsilon_k}\, \mathcal{F}_k$ is given by expanding $\mathcal{F}_k$ around its center w^k by a factor of $\sqrt{1 + \epsilon_k}$. Therefore, $\log \mathrm{vol}\, \mathrm{MVCE}(\mathcal{S}) \leq \log \mathrm{vol}\, \mathcal{F}_k + (n/2) \log(1 + \epsilon_k)$, which proves the second inequality in (22).

We follow an argument similar to that in the proof of Theorem 4.1 to establish the first inequality (cf. (20), (21), and (19)). At step k of Algorithm 4.1, $u^k \in \mathbb{R}^{2n+k}$ is a feasible solution of the optimization problem $(\mathbf{D}(\mathcal{X}_\mathbf{k}))$. Therefore,

$$\log \mathrm{vol}\, \mathcal{F}_k = \log \eta + \frac{n}{2} \log n + \frac{1}{2} \log \det \Pi_k(u^k),$$
$$\leq \log \eta + \frac{n}{2} \log n + \frac{1}{2} \log \det \Pi_k(u_*^k),$$
$$= \log \mathrm{vol}\, \mathrm{MVCE}(\mathcal{X}_k),$$

where u_*^k denotes the optimal solution of $(\mathbf{D}(\mathcal{X}_\mathbf{k}))$ and $\Pi_k : \mathbb{R}^{2n+k} \to \mathcal{S}^{n+1}$ is a linear operator given by

$$(23) \qquad \Pi_k(u) := \sum_{j=1}^{2n+k} u_j \left[\begin{array}{cc} x^j (x^j)^T & x^j \\ (x^j)^T & 1 \end{array} \right].$$

Since $\mathcal{X}_k \subset \mathcal{S}$, the first inequality follows. $\qquad \square$

The following corollary immediately follows from Lemma 4.2.

COROLLARY 4.3. *For any $k = 0, 1, 2, \ldots$, $\epsilon_k \geq 0$. Furthermore, if Algorithm 4.1 does not terminate at step k, then $\epsilon_k > (1 + \epsilon)^{2/n} - 1$.*

So far, we have established the following results: (i) $\mathrm{vol}\, \mathcal{F}_0$ is a provable approximation to $\mathrm{vol}\, \mathrm{MVCE}(\mathcal{S})$ and (ii) the sequence of ellipsoids $\mathcal{F}_k$ generated by Algorithm 4.1 yields a sequence of lower bounds on $\mathrm{vol}\, \mathrm{MVCE}(\mathcal{S})$. Our next goal is to demonstrate that $\{\mathrm{vol}\, \mathcal{F}_k\}, k = 0, 1, \ldots$, is a strictly increasing sequence, which implies that Algorithm 4.1 produces increasingly sharper lower bounds to $\mathrm{vol}\, \mathrm{MVCE}(\mathcal{S})$. At this stage, it is worth noticing that the line search problem $\mathrm{LS}(k)$ precisely computes the next trial ellipsoid which yields the largest increase in the volume for the particular updating scheme of Algorithm 4.1.

PROPOSITION 4.4. *For any $k = 0, 1, 2, \ldots$,*

$$(24) \qquad \log \mathrm{vol}\, \mathcal{F}_{k+1} \geq \log \mathrm{vol}\, \mathcal{F}_k + \left\{ \begin{array}{ll} \frac{1}{2} \log 2 - \frac{1}{4} > 0 & \text{if } \epsilon_k \geq \frac{n+1}{n}, \\[2ex] \frac{1}{16} \epsilon_k^2 & \text{if } \epsilon_k < \frac{n+1}{n}. \end{array} \right.$$

Proof. Our proof mimics Khachiyan's argument [15]. By the definition of ϵ_k, we have $1 + \epsilon_k = (x^{2n+k+1} - w^k)^T M^k (x^{2n+k+1} - w^k)$. Let $z^k := x^{2n+k+1} - w^k$. It follows from (15) that

$$
\begin{aligned}
\log \det(M^{k+1})^{-1} &= \log \det \left\{ (1 - \beta_k) \left[(M^k)^{-1} + n\beta_k z^k (z^k)^T \right] \right\}, \\
&= n \log(1 - \beta_k) + \log \det \left[(M^k)^{-1} \left(I + n\beta_k M^k z^k (z^k)^T \right) \right], \\
&= \log \det(M^k)^{-1} + n \log(1 - \beta_k) + \log \left[1 + n\beta_k (1 + \epsilon_k) \right], \\
&= \log \det(M^k)^{-1} - n \log \left(1 + \frac{\beta_k}{1 - \beta_k} \right) + \log \left(1 + \left(\frac{n}{n+1} \right) \epsilon_k \right), \\
&= \log \det(M^k)^{-1} - n \log \left(1 + \frac{\epsilon_k}{(n+1)(1+\epsilon_k) - \epsilon_k} \right) \\
&\quad + \log \left(1 + \left(\frac{n}{n+1} \right) \epsilon_k \right), \\
&\geq \log \det(M^k)^{-1} - \frac{\left(\frac{n}{n+1} \right) \epsilon_k}{1 + \left(\frac{n}{n+1} \right) \epsilon_k} + \log \left(1 + \left(\frac{n}{n+1} \right) \epsilon_k \right),
\end{aligned}
$$

where we used the definition of β_k in the last two equalities and the inequality $\log(1 + \zeta) \leq \zeta$ for $\zeta > -1$. Since $\log \operatorname{vol} \mathcal{F}_k = \log \eta + \log \det(M^k)^{-1/2} = \log \eta + (1/2) \log \det(M^k)^{-1}$, it follows that

$$
\log \operatorname{vol} \mathcal{F}_{k+1} \geq \log \operatorname{vol} \mathcal{F}_k + \frac{1}{2} \log \left(1 + \left(\frac{n}{n+1} \right) \epsilon_k \right) - \frac{\left(\frac{n}{n+1} \right) \epsilon_k}{2 \left(1 + \left(\frac{n}{n+1} \right) \epsilon_k \right)}.
$$

The assertion follows from the observation that $f(\nu) := (1/2) \log(1+\nu) - \nu/[2(1+\nu)]$ is a strictly increasing function for $\nu \geq 0$ and $f(\nu) \geq \nu^2/16$ for $\nu \in [0, 1]$. $\square$

We are now ready to analyze the iteration complexity of Algorithm 4.1. To this end, we define the following parameters:

$$
(25) \qquad \tau_\rho := \min \left\{ k : \left(\frac{n}{n+1} \right) \epsilon_k \leq 1/2^\rho \right\}, \quad \rho = 0, 1, 2, \ldots.
$$

The next lemma establishes certain properties of τ_ρ.

LEMMA 4.5. τ_ρ *satisfies the following relationships:*

$$
(26) \qquad \tau_0 = O(n \log n),
$$

$$
(27) \qquad \tau_\rho - \tau_{\rho-1} \leq n 2^{\rho+5}, \quad \rho = 1, 2, \ldots.
$$

Proof. By Theorem 4.1, $\log \operatorname{vol} \mathcal{F}_0 \leq \log \operatorname{vol} \operatorname{MVCE}(\mathcal{S}) \leq \log \operatorname{vol} \mathcal{F}_0 + 2n \log n + (n/2) \log 2$. At every iteration k with $\epsilon_k > (n+1)/n$, we have $\log \operatorname{vol} \mathcal{F}_{k+1} - \log \operatorname{vol} \mathcal{F}_k \geq (1/2) \log 2 - 1/4 > 0$ by Proposition 4.4. Therefore, $\tau_0 = O(n \log n)$.

Let us now consider $\tau_\rho - \tau_{\rho-1}$, $\rho \geq 1$. For simplicity, let $\gamma := \tau_{\rho-1}$. By definition of $\tau_{\rho-1}$, it follows from Lemma 4.2 that $\log \operatorname{vol} \mathcal{F}_\gamma \leq \log \operatorname{vol} \operatorname{MVCE}(\mathcal{S}) \leq \log \operatorname{vol} \mathcal{F}_\gamma + (n/2) \log(1 + [(n+1)/n]2^{-(\rho-1)}) \leq \log \operatorname{vol} \mathcal{F}_\gamma + (n+1)2^{-\rho}$. By Proposition 4.4, $\log \operatorname{vol} \mathcal{F}_{k+1} - \log \operatorname{vol} \mathcal{F}_k \geq [(n+1)/n]^2 2^{-(2\rho+4)} \geq 2^{-(2\rho+4)}$ at every iteration k with $\epsilon_k > [(n+1)/n]2^{-\rho}$. Therefore, $\tau_\rho - \tau_{\rho-1} \leq [(n+1)2^{-\rho}]/2^{-(2\rho+4)} = (n+1)2^{\rho+4} \leq n 2^{\rho+5}$, which completes the proof. $\square$

Lemma 4.5 enables us to establish the following result.

LEMMA 4.6. *Let* $\mu \in (0,1)$. *Algorithm* 4.1 *computes an iterate with* $\epsilon_k \leq \mu$ *in* $O(n(\log n + \mu^{-1}))$ *iterations.*

Proof. Let σ be a positive integer such that $[(n+1)/n]2^{-\sigma} \leq \mu \leq [(n+1)/n]2^{1-\sigma}$. Therefore, after $k = \tau_\sigma$ iterations, we already have $\epsilon_k \leq [(n+1)/n]2^{-\sigma} \leq \mu$. However,

$$\tau_\sigma = \tau_0 + \sum_{\rho=1}^{\sigma}(\tau_\rho - \tau_{\rho-1}) \leq \tau_0 + 64n\sum_{\rho=1}^{\sigma}2^{\rho-1} \leq \tau_0 + 64n2^\sigma \leq O\left(n\log n + \frac{n}{\mu}\right),$$

where we used Lemma 4.5 and the inequality $2^\sigma \leq 4/\mu$. $\quad\square$

We are now in a position to establish the iteration complexity of Algorithm 4.1.

THEOREM 4.7. *Let* $\epsilon > 0$. *Algorithm* 4.1 *computes a* $(1+\epsilon)$-*approximation to* MVCE($\mathcal{S}$) *after at most* $O(n(\log n + [(1+\epsilon)^{2/n} - 1]^{-1}))$ *iterations.*

Proof. We first establish that Algorithm 4.1 returns a $(1+\epsilon)$-approximation to MVCE($\mathcal{S}$) upon termination. Let κ denote the index of the final iterate. We have $\epsilon_\kappa \leq (1+\epsilon)^{2/n} - 1$. The trial ellipsoid $\mathcal{F}_\kappa$ satisfies $\mathcal{S} \subseteq \sqrt{1+\epsilon_\kappa}\,\mathcal{F}_\kappa$, which together with Lemma 4.2 implies that

$$\text{vol } \mathcal{F}_\kappa \leq \text{vol MVCE}(\mathcal{S}) \leq \text{vol}\sqrt{1+\epsilon_\kappa}\mathcal{F}_\kappa = (1+\epsilon_\kappa)^{n/2}\text{vol } \mathcal{F}_\kappa \leq (1+\epsilon)\text{vol } \mathcal{F}_\kappa.$$

Therefore, $\sqrt{1+\epsilon_\kappa}\,\mathcal{F}_\kappa$ is indeed a $(1+\epsilon)$-approximation to MVCE($\mathcal{S}$).

We now prove the iteration complexity. If $\epsilon \geq [2+(1/n)]^{n/2} - 1$, then $(1+\epsilon)^{2/n} - 1 \geq (n+1)/n$, which implies that at most $\tau_0 = O(n\log n)$ iterations already suffice. Otherwise, the result follows from Lemma 4.6. $\quad\square$

Remark 1. The iteration complexity of Algorithm 4.1 is asymptotically identical to that of the algorithm of Kumar and Yıldırım [18] that computes a $(1+\epsilon)$-approximation to the MVCE of a finite set of m points.

We now establish the overall complexity of Algorithm 4.1.

THEOREM 4.8. *Algorithm* 4.1 *computes a* $(1+\epsilon)$-*approximation to MVCE($\mathcal{S}$) in*

$$O\left(mn^{O(1)}(\log n + [(1+\epsilon)^{2/n} - 1]^{-1})\right)$$

operations, where $O(1)$ *denotes a universal constant greater than four.*

Proof. We already have the iteration complexity from Theorem 4.7. We need only analyze the computational cost of each iteration.

Let us start with the initialization stage. By Lemma 3.1, Algorithm 3.1 runs in $O(mn^3)$ operations. w^0 and $(M^0)^{-1}$ can be computed in $O(n^2)$ and $O(n^3)$ operations, respectively. The furthest point x^{2n+1} from the center of $\mathcal{F}_0$ can be determined by solving m separate quadratic optimization problems with a single ellipsoidal constraint. By Proposition 2.6, each optimization problem can be solved in $O(n^{O(1)} + n^3)$ operations. Finally, it takes $O(n^2)$ operations to compute ϵ_0. Therefore, the overall complexity of the initialization step is $O(m(n^{O(1)} + n^3))$ operations. Similarly, at iteration k, the major work is the computation of the furthest point x^{2n+k+1}, which can be performed in $O(m(n^{O(1)} + n^3))$ operations. Therefore, the overall running time of Algorithm 4.1 is given by $O(mn^{O(1)}(\log n + [(1+\epsilon)^{2/n} - 1]^{-1}))$ operations. $\quad\square$

Remark 2. The overall complexity of Algorithm 4.1 is linear in m, the number of ellipsoids. This suggests that, in theory, Algorithm 4.1 is especially well-suited for instances of the MVCE problem that satisfy $m \gg n$ and for moderate values of ϵ. In addition, if $\epsilon \in (0,1)$, we have $(1+\epsilon)^{2/n} - 1 = \Theta(\epsilon/n)$, in which case the running time of Algorithm 4.1 can be simplified to $O((1/\epsilon)mn^{O(1)})$ operations, where $O(1)$ is now

a universal constant greater than five. Note that the running time of Algorithm 4.1 is polynomial for fixed ϵ.

We close this section by establishing that the convex hull of the finite set of points collected by Algorithm 4.1 serves as a reasonably good approximation to $\mathcal{S}$ in the sense that their respective MVCEs are closely related.

PROPOSITION 4.9. *Let κ denote the index of the final iterate of Algorithm* 4.1. *Then, $\mathcal{X}_\kappa$ satisfies*

$$(28) \qquad \text{vol } MVCE(\mathcal{X}_\kappa) \leq \text{vol } MVCE(\mathcal{S}) \leq (1+\epsilon)\text{vol } MVCE(\mathcal{X}_\kappa).$$

In addition,

$$(29) \qquad |\mathcal{X}_\kappa| = O\left(n(\log n + [(1+\epsilon)^{2/n} - 1]^{-1})\right).$$

Proof. We first prove (28). Note that the first inequality is obvious since $\mathcal{X}_\kappa \subset \mathcal{S}$. The second inequality follows from the relationships vol $\mathcal{F}_\kappa \leq$ vol MVCE$(\mathcal{X}_\kappa) \leq$ vol MVCE$(\mathcal{S})$ (see the proof of Lemma 4.2) and vol MVCE$(\mathcal{S}) \leq (1+\epsilon)$vol $\mathcal{F}_\kappa$ (see the proof of Theorem 4.7).

Since $|\mathcal{X}_\kappa| = 2n + \kappa$, (29) simply follows from Theorem 4.7.   □

Remark 3. Proposition 4.9 establishes that Algorithm 4.1 computes a finite set of points $\mathcal{X}_\kappa \subset \mathcal{S}$ whose MVCE is related to MVCE$(\mathcal{S})$ via (28). In addition, $|\mathcal{X}_\kappa|$ depends only on the dimension n and the approximation factor ϵ but is independent of the number of ellipsoids m. Furthermore, for $\epsilon \in (0,1)$, we have $|\mathcal{X}_\kappa| = O(n^2/\epsilon)$. Therefore, $\mathcal{X}_\kappa$ serves as a finite core set for $\mathcal{S}$. Viewed from this perspective, Proposition 4.9 is an addition to the previous core set results for other geometric optimization problems [17, 8, 7, 9, 1, 18].

Remark 4. In [18], a similar core set result has been established for the MVCE problem for a finite set of m points in $\mathbb{R}^n$. It is remarkable that asymptotically the same result holds regardless of the difference in the underlying geometric structures of the two input sets. In particular, the main ingredient in [18] that leads to the improved complexity result over Khachiyan's algorithm [15] as well as the core set result is the initial volume approximation. In a similar manner, the counterpart of this initialization stage (cf. Algorithm 3.1) enables us to extend the algorithm of Kumar and Yıldırım to a set of ellipsoids. Khachiyan's algorithm cannot be extended to a set of ellipsoids as it relies on the finiteness property of the input set at the initialization stage.

5. Rounding. In this section, we establish that Algorithm 4.1 can also be used to compute a $(1+\delta)n$-rounding of $\mathcal{S} := \text{conv}(\cup_{i=1}^m \mathcal{E}_i)$, where $\mathcal{E}_1, \ldots, \mathcal{E}_m \subset \mathbb{R}^n$ are full-dimensional ellipsoids and $\delta > 0$. We assume that $\mathcal{S}$ is not symmetric around the origin.

Our analysis closely follows Khachiyan's treatment for an input set of a finite number of points in $\mathbb{R}^n$ [15]. At iteration k of Algorithm 4.1, let $q^j := [(x^j)^T, 1]^T \in \mathbb{R}^{n+1}$, $j = 1, \ldots, 2n+k$, and let $\mathcal{Q}_k := \text{conv}(\{\pm q^1, \ldots, \pm q^{2n+k}\})$, which is a centrally symmetric polytope in $\mathbb{R}^{n+1}$ (i.e., $\mathcal{Q}_k = -\mathcal{Q}_k$).

Let $u^k \in \mathbb{R}^{2n+k}$ denote the iterate at iteration k of Algorithm 4.1. Let us define a full-dimensional ellipsoid $\mathcal{G}_k \subset \mathbb{R}^{n+1}$ given by

$$(30) \qquad \mathcal{G}_k := \{y \in \mathbb{R}^{n+1} : y^T \Pi_k(u^k)^{-1} y \leq 1\},$$

where $\Pi_k : \mathbb{R}^{2n+k} \to \mathcal{S}^{n+1}$ is a linear operator defined by (23). Since u^k is a feasible solution of $(\mathbf{D}(\mathcal{X}_\mathbf{k}))$, it follows from [15, Lemma 2] that $\mathcal{G}_k \subseteq \mathcal{Q}_k$. Furthermore, for any q^j, $j = 1, \ldots, 2n+k$, we have

$$(q^j)^T \Pi_k (u^k)^{-1} q^j = \begin{bmatrix} (x^j)^T & 1 \end{bmatrix} \begin{bmatrix} I & 0 \\ -(w^k)^T & 1 \end{bmatrix} \begin{bmatrix} nM_k & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} I & -w^k \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x^j \\ 1 \end{bmatrix},$$

$$= n(x^j - w^k)^T M_k (x^j - w^k) + 1,$$

$$\leq n(1 + \epsilon_k) + 1,$$

which, together with the previous inclusion, implies that

$$(31) \qquad \mathcal{G}_k \subseteq \mathcal{Q}_k \subseteq \sqrt{1 + n(1 + \epsilon_k)} \, \mathcal{G}_k;$$

i.e., $\sqrt{1 + n(1 + \epsilon_k)} \, \mathcal{G}_k$ is a $\sqrt{(1 + \tilde{\delta})(n + 1)}$-rounding of $\mathcal{Q}_k$, where $\tilde{\delta} := (n\epsilon_k)/(n+1)$.

Let $\mathcal{H}_k := \{x \in \mathbb{R}^n : [x^T \quad 1]^T \in \sqrt{1 + n(1 + \epsilon_k)} \, \mathcal{G}_k \cap \Lambda\}$, where

$$(32) \qquad \Lambda := \{y \in \mathbb{R}^{n+1} : y_{n+1} = 1\}.$$

Note that $\mathcal{H}_k \subset \mathbb{R}^n$ is a full-dimensional ellipsoid. By [15, Lemma 5],

$$(33) \qquad \frac{1}{(1 + \epsilon_k)n} \, \mathcal{H}_k \subseteq \mathrm{conv}(\mathcal{X}_k) \subseteq \mathcal{H}_k;$$

i.e., $\mathcal{H}_k \subset \mathbb{R}^n$ is a $(1 + \epsilon_k)n$-rounding of $\mathrm{conv}(\mathcal{X}_k)$. However, it is straightforward to verify that $x \in \mathcal{H}_k$ if and only if $(x - w^k)^T M^k (x - w^k) \leq 1 + \epsilon_k$, which implies that $\mathcal{H}_k = \sqrt{1 + \epsilon_k} \, \mathcal{F}_k$. Since $\mathrm{conv}(\mathcal{X}_k) \subseteq \mathcal{S} \subseteq \sqrt{1 + \epsilon_k} \, \mathcal{F}_k$, it follows from (33) that

$$(34) \qquad \frac{1}{(1 + \epsilon_k)n} \, \mathcal{H}_k \subseteq \mathrm{conv}(\mathcal{X}_k) \subseteq \mathcal{S} \subseteq \mathcal{H}_k;$$

i.e., $\mathcal{H}_k$ is simultaneously a $(1 + \epsilon_k)n$-rounding of $\mathrm{conv}(\mathcal{X}_k)$ and of $\mathcal{S}$. Therefore, in order to obtain a $(1 + \delta)n$-rounding of $\mathcal{S}$, it suffices to run Algorithm 4.1 until $\epsilon_k \leq \delta$. We summarize this result in the following corollary, whose proof follows directly from Lemma 4.6.

COROLLARY 5.1. *Given $\delta > 0$, Algorithm* 4.1 *computes a $(1 + \delta)n$-rounding of $\mathcal{S}$ in*

$$O(mn^{O(1)}(\log n + \delta^{-1}))$$

arithmetic operations, where $O(1)$ is a universal constant greater than four. In addition, upon termination of Algorithm 4.1*, the ellipsoid computed by Algorithm* 4.1 *is also a $(1 + \delta)n$-rounding of the convex hull of a finite subset $\mathcal{X}_k \subset \mathcal{S}$ with the property that*

$$(35) \qquad |\mathcal{X}_k| = O\left(n(\log n + \delta^{-1})\right).$$

Remark 5. Upon termination of Algorithm 4.1 with a $(1 + \delta)n$-rounding of $\mathcal{S}$, Corollary 5.1 establishes that $\mathcal{X}_k$ is an $\tilde{\epsilon}$-core set for $\mathcal{S}$, where $\tilde{\epsilon} := (1 + \delta)^{n/2} - 1$. In fact, Khachiyan's algorithm [15] is motivated by first computing a $(1 + \delta)n$-rounding of $\mathcal{S}$ and then choosing δ in such a way that the ellipsoid computed by the algorithm is a $(1 + \epsilon)$-approximation to the MVCE.

We close this section by noting that Corollary 5.1 can be improved if $\mathcal{S}$ is centrally symmetric. In this case, we no longer need to "lift" the vectors in $\mathcal{X}_k$ to $\mathbb{R}^{n+1}$. A similar argument can be invoked to establish that $\mathcal{H}_k := \sqrt{1 + \epsilon_k} \, \mathcal{F}_k$ satisfies

$$\frac{1}{\sqrt{(1 + \epsilon_k)n}} \, \mathcal{H}_k \subseteq \mathrm{conv}(\{\pm x^1, \ldots, \pm x^{2n+k}\}) \subseteq \mathcal{S} \subseteq \mathcal{H}_k.$$

6. Extensions to other sets. In this section, we discuss the extent to which Algorithm 4.1 can be used to compute an approximate MVCE and an approximate n-rounding of other input sets. Let $\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_m$ denote m objects in $\mathbb{R}^n$ and let $\mathcal{T} := \cup_{i=1}^m \mathcal{T}_i$. In order to extend Algorithm 4.1, we identify the following two subroutines that need to be implemented efficiently:

  1. *Subroutine* 1: Optimizing a linear function over $\mathcal{T}$.
  2. *Subroutine* 2: Maximizing a quadratic function over $\mathcal{T}$.

Note that Subroutine 1 is required by Algorithm 3.1. Similarly, the computation of the furthest point from the center of a trial ellipsoid (in its ellipsoidal norm) is equivalent to Subroutine 2. All of the other operations of Algorithm 4.1 can be performed efficiently for any input set $\mathcal{T}$.

Let us now consider specific examples of input sets. Clearly, a finite set of points and a finite set of balls would be special cases of a finite set of ellipsoids. Therefore, Algorithm 4.1 would be trivially applicable in these cases. We just remark that certain subroutines can be implemented more efficiently for these input sets. For a finite set of points, Subroutines 1 and 2 can be performed in $O(mn)$ and $O(mn^2)$ operations, respectively. We then recover the algorithm of Kumar and Yıldırım [18]. For a finite set of balls, while Subroutine 1 can still be implemented in $O(mn)$ operations, Subroutine 2 would require the same computational cost as that required by a set of ellipsoids. Therefore, the running time of Algorithm 4.1 would asymptotically remain the same for a finite set of balls. A similar argument also holds for an input set of ellipsoids, each of which is defined by the same matrix $Q = Q^i$, $i = 1, \ldots, m$, since such an input set can be a priori transformed into a set of balls.

6.1. Set of half ellipsoids. Let $\mathcal{T}_i := \{x \in \mathbb{R}^n : (x - c^i)^T Q^i (x - c^i) \leq 1, (f^i)^T x \leq \alpha^i\}$, where $c^i \in \mathbb{R}^n$ and $Q^i \in \mathcal{S}^n$ are positive definite, $f^i \in \mathbb{R}^n$ with $f^i \neq 0$, and $\alpha^i \in \mathbb{R}$, $i = 1, \ldots, m$. $\mathcal{T}_i$ is simply given by the intersection of a full-dimensional ellipsoid and a half-space. We will refer to each $\mathcal{T}_i$ as a half-ellipsoid. To avoid trivialities, we assume that each $\mathcal{T}_i$ has a nonempty interior. It follows from the results of Sturm and Zhang [28] that the problem of optimizing any quadratic (hence linear) objective function over $\mathcal{T}_i$ can be cast as an equivalent SDP problem with a fixed number of constraints using a technique similar to that used in the proof of Proposition 2.6. Since both Subroutines 1 and 2 naturally decompose into a linear and quadratic optimization problem over each $\mathcal{T}_i$, respectively, it follows from Corollary 2.5 that both of them can be implemented in polynomial time. Therefore, Algorithm 4.1 can compute an approximate MVCE and an approximate n-rounding of a set of half-ellipsoids in polynomial time.

6.2. Set of intersections of a pair of ellipsoids. Let $\mathcal{T}_i := \{x \in \mathbb{R}^n : (x - c^i)^T Q^i (x - c^i) \leq 1, (x - h^i)^T Q^i (x - h^i) \leq 1\}$, where $c^i \in \mathbb{R}^n$, $h^i \in \mathbb{R}^n$, and $Q^i \in \mathcal{S}^n$ are positive definite, $i = 1, \ldots, m$. Note that each $\mathcal{T}_i$ is given by the intersection of two ellipsoids defined by the same matrix Q^i with different centers. Similarly to the previous case, Sturm and Zhang [28] establish that the problem of optimizing any quadratic (hence linear) objective function over $\mathcal{T}_i$ can be decomposed into two quadratic (linear) optimization problems over a half-ellipsoid, each of which can be solved in polynomial time. Therefore, Algorithm 4.1 can compute an approximate MVCE and an approximate n-rounding of a set of intersections of a pair of ellipsoids in polynomial time. We remark that the complexity of solving a general quadratic optimization problem over the intersection of two arbitrary ellipsoids is still an open problem.

6.3. Other sets and limitations. Based on the previous examples, it is clear that Algorithm 4.1 can be applied to any input set as long as Subroutines 1 and 2 admit efficient implementations. While Subroutine 1 can be performed efficiently for a rather large class of input sets (e.g., classes of convex sets that admit efficiently computable barrier functions [19]), Subroutine 2 can be efficiently implemented only in very special cases, some of which have been outlined in this section.

For instance, if $\mathcal{T}$ is given by the union of polytopes $\mathcal{T}_i := \{x \in \mathbb{R}^n : A^i x \leq b^i\}$, where $A^i \in \mathbb{R}^{m \times n}$ and $b^i \in \mathbb{R}^m$, $i = 1, \ldots, m$, then Subroutine 1 reduces to linear programming, which can be solved efficiently using interior-point methods combined with a finite termination strategy [35]. However, maximizing a convex quadratic function over a polytope is in general an NP-hard problem. Therefore, even in the case of a single polytope defined by linear inequalities, the problem of computing an approximate MVCE is computationally intractable. We remark that the maximum volume inscribed ellipsoid in a polytope defined by linear inequalities can be efficiently approximated (see, e.g., [16]).

In summary, the extent to which Algorithm 4.1 can be applied to other input sets is largely determined by whether Subroutine 2 can be implemented efficiently. Since quadratic optimization over various feasible regions is an active area of research [28, 36], further progress in establishing polynomial complexity may widen the domain of input sets to which Algorithm 4.1 can be applied.

7. Concluding remarks. In this paper, we established that the first-order algorithm of Kumar and Yıldırım [18] that computes an approximate MVCE of a finite set of points can be extended to compute the MVCE of the union of finitely many full-dimensional ellipsoids without compromising the polynomial-time complexity for a fixed approximation parameter $\epsilon > 0$. Moreover, the iteration complexity of our extension and the core set size remain asymptotically identical. In addition, we establish that our algorithm can also compute an appproximate n-rounding of the convex hull of a finite number of ellipsoids. We discuss how the framework of our algorithm can be extended to compute an approximate MVCE and an approximate n-rounding of other input sets in polynomial time and present certain limitations. Our core set result is an addition to the recent sequence of works on core sets for several geometric optimization problems [17, 8, 7, 9, 1, 18].

While our algorithm has a polynomial-time complexity in theory, it would be especially well suited for instances of the MVCE problem with $m \gg n$ and moderately small values of ϵ. In particular, our algorithm would be applicable to the problem of constructing a bounding volume hierarchy as the objects lie in three-dimensional space (i.e., $n = 3$) and a fixed parameter ϵ usually suffices for practical applications. To the best of our knowledge, this is the first result in the literature towards approximating the convex hull of a union of ellipsoids by that of a finite subset whose size depends on only the dimension n and the parameter ϵ.

On the other hand, our algorithm would probably not be practical if a higher accuracy (i.e., a smaller ϵ) were required or if the dimension n were large. In addition, it is well known that first-order algorithms in general suffer from slow convergence in practice, especially for smaller values of ϵ. Our preliminary computational results indicate that both of the first-order algorithms of [15, 18] for an input set of points tend to take an excessive number of iterations as ϵ is decreased, which suggests that the practical performance of these algorithms is indeed closely related to the worst-case theoretical complexity bounds. Motivated by the core set result established in this paper and the encouraging computational results based on a similar core set

result for the minimum enclosing ball problem [17], we intend to work on a column generation algorithm for the MVCE problem with an emphasis on establishing an upper bound on the number of subproblems solved to obtain a desired accuracy.

Very recently, Todd and Yıldırım [30] proposed a modification of the algorithm of Kumar and Yıldırım [18] that computes an approximate MVCE and an approximate n-rounding of a finite set of points. Their modification allows "dropping" points from a working core set throughout the algorithm and maintains the same complexity bound as that of the algorithm of [18]. As such, it has the potential of computing smaller core sets in practice. We remark that the same idea can easily be incorporated into our algorithm for a set of ellipsoids without any increase in the asymptotic complexity bound.

REFERENCES

[1] P. K. AGARWAL, R. POREDDY, K. R. VARADARAJAN, AND H. YU, *Practical methods for shape fitting and kinetic data structures using core sets*, in Proceedings of the 20th Annual ACM Symposium on Computational Geometry, 2004.

[2] A. BEN-TAL AND A. NEMIROVSKI, *Lectures on Modern Convex Optimization: Analysis, Algorithms, and Engineering Applications*, MPS/SIAM Ser. Optim. 2, SIAM, Philadelphia, 2001.

[3] U. BETKE AND M. HENK, *Approximating the volume of convex bodies*, Discrete Comput. Geom., 10 (1993), pp. 15–21.

[4] L. BLUM, M. SHUB, AND S. SMALE, *On a theory of computation and complexity over the real numbers: NP-completeness, recursive functions, and universal machines*, Bull. Amer. Math. Soc. (N.S.), 21 (1989), pp. 1–46.

[5] S. BOYD, L. EL GHAOUI, E. FERON, AND V. BALAKRISHNAN, *Linear Matrix Inequalities in System and Control Theory*, SIAM Stud. Appl. Math. 15, SIAM, Philadelphia, 1994.

[6] S. BOYD AND L. VANDENBERGHE, *Convex Optimization*, Cambridge University Press, Cambridge, UK, 2004.

[7] M. BĂDOIU AND K. L. CLARKSON, *Smaller core-sets for balls*, in Proceedings of the 14th ACM-SIAM Symposium on Discrete Algorithms, 2003, Baltimore, MD, pp. 801–802.

[8] M. BĂDOIU, S. HAR-PELED, AND P. INDYK, *Approximate clustering via core-sets*, in Proceedings of the 34th Annual ACM Symposium on Theory of Computing, 2002, pp. 250–257.

[9] T. M. CHAN, *Faster core-set constructions and data-stream algorithms in fixed dimensions*, Comput. Geom., 35 (2006), pp. 20–35.

[10] Y. CHIEN AND J. LIU, *Improvements to ellipsoidal fit based collision detection*, Tech. report TR-IIS-03-001, Academia Sinica, Institute of Information Science, Taiwan, Taipei, Republic of China, 2003.

[11] J. H. CLARK, *Hierarchical geometric models for visible surface algorithms*, Commun. ACM, 19 (1976), pp. 547–554.

[12] M. GRÖTSCHEL, L. LOVÁSZ, AND A. SCHRIJVER, *Geometric Algorithms and Combinatorial Optimization*, Springer, New York, 1988.

[13] F. JOHN, *Extremum problems with inequalities as subsidiary conditions*, in Studies and Essays Presented to R. Courant on his 60th Birthday, January 8, 1948, Interscience, New York, 1948, pp. 187–204; reprinted in *Fritz John, Collected Papers*, Vol. 2, J. Moser, ed., Birkhäuser Boston, Boston, 1985, pp. 543–560.

[14] M. JU, J. LIU, S. SHIANG, Y. CHIEN, K. HWANG, AND W. LEE, *Fast and accurate collision detection based on enclosed ellipsoids*, Robotica, 19 (2001), pp. 381–394.

[15] L. G. KHACHIYAN, *Rounding of polytopes in the real number model of computation*, Math. Oper. Res., 21 (1996), pp. 307–320.

[16] L. G. KHACHIYAN AND M. J. TODD, *On the complexity of approximating the maximal inscribed ellipsoid for a polytope*, Math. Programming, 61 (1993), pp. 137–159.

[17] P. KUMAR, J. S. B. MITCHELL, AND E. A. YILDIRIM, *Approximate minimum enclosing balls in high dimensions using core-sets*, ACM J. Exp. Algorithmics, 8 (2003), 29 pp.

[18] P. KUMAR AND E. A. YILDIRIM, *Minimum volume enclosing ellipsoids and core sets*, J. Optim. Theory Appl., 126 (2005), pp. 1–21.

[19] YU. NESTEROV AND A. NEMIROVSKII, *Interior Point Polynomial Algorithms in Convex Programming*, SIAM Stud. Appl. Math. 13, SIAM, Philadelphia, 1994.

[20] I. PÓLIK AND T. TERLAKY, *A comprehensive study of the $\mathcal{S}$-lemma*, Tech. report, McMaster University, Advanced Optimization Laboratory, Hamilton, ON, Canada, 2004.

[21] L. PORKOLAB AND L. KHACHIYAN, *On the complexity of semidefinite programs*, J. Global Optim., 10 (1997), pp. 351–365.

[22] M. V. RAMANA, *An exact duality theory for semidefinite programming and its complexity implications*, Math. Programming, 77 (1997), pp. 129–162.

[23] F. RENDL AND H. WOLKOWICZ, *A semidefinite framework for trust region subproblems with applications to large scale minimization*, Math. Programming, 77 (1997), pp. 273–299.

[24] J. RENEGAR, *On the computational complexity and geometry of the first-order theory of the reals, Part* I: *Introduction; preliminaries; the geometry of the semi-algebraic sets; the decision problem for the existential theory of the reals*, J. Symbolic Comput., 13 (1992), pp. 255–299.

[25] E. RIMON AND S. BOYD, *Obstacle collision detection using best ellipsoid fit*, J. Intell. Robot. Syst., 18 (1997), pp. 105–126.

[26] S. SHIANG, J. LIU, AND Y. CHIEN, *Estimate of minimum distance between convex polyhedra based on enclosed ellipsoids*, in Proceedings of the 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2000, pp. 739–744.

[27] R. J. STERN AND H. WOLKOWICZ, *Indefinite trust region subproblems and nonsymmetric eigenvalue perturbations*, SIAM J. Optim., 5 (1995), pp. 286–313.

[28] J. F. STURM AND S. Z. ZHANG, *On cones of nonnegative quadratic functions*, Math. Oper. Res., 28 (2003), pp. 246–267.

[29] P. SUN AND R. M. FREUND, *Computation of minimum volume covering ellipsoids*, Oper. Res., 52 (2004), pp. 690–706.

[30] M. J. TODD AND E. A. YILDIRIM, *On Khachiyan's algorithm for the computation of minimum volume enclosing ellipsoids*, Tech. report TR 1435, School of Operations Research and Industrial Engineering, Cornell University, Ithaca, NY, 2005.

[31] K. C. TOH, *Primal-dual path-following algorithms for determinant maximization problems with linear matrix inequalities*, Comput. Optim. Appl., 14 (1999), pp. 309–330.

[32] L. VANDENBERGHE, S. BOYD, AND S.-P. WU, *Determinant maximization with linear matrix inequality constraints*, SIAM J. Matrix Anal. Appl., 19 (1998), pp. 499–533.

[33] V. A. YAKUBOVICH, *$\mathcal{S}$-procedure in nonlinear control theory*, Vestn. Leningr. Univ., 4 (1977), pp. 73–93 (in English).

[34] Y. YE, *On affine scaling algorithms for nonconvex quadratic programming*, Math. Programming, 56 (1992), pp. 285–300.

[35] Y. YE, *On the finite convergence of interior-point algorithms for linear programming*, Math. Programming, 57 (1992), pp. 325–335.

[36] Y. YE AND S. ZHANG, *New results on quadratic minimization*, SIAM J. Optim., 14 (2003), pp. 245–267.