Chapter 3

The Ellipsoid Method

In 1979 a note of L. G. Khachiyan indicated how an algorithm, the so-called **ellipsoid method**, originally devised for nonlinear nondifferentiable optimization, can be modified in order to check the feasibility of a system of linear inequalities in polynomial time. This result caused great excitement in the world of mathematical programming since it implies the polynomial time solvability of linear programming problems.

This excitement had several causes. First of all, many researchers all over the world had worked hard on the problem of finding a polynomial time algorithm for linear programming for a long time without success. So a really major open problem had been solved.

Secondly, many people believe that $\mathcal{P} = \mathcal{NP} \cap \text{co-}\mathcal{NP}$ – cf. Section 1.1 – and the linear programming problem was one of the few problems known to belong to $\mathcal{NP} \cap \text{co-}\mathcal{NP}$ but that had not been shown to be in $\mathcal{P}$. Thus, a further indication for the correctness of this conjecture was obtained.

Thirdly, the ellipsoid method together with the additional number theoretical "tricks" was so different from all the algorithms for linear programming considered so far that the method itself and the correctness proof were a real surprise.

Fourthly, the ellipsoid method, although "theoretically efficient", did not prove to be "practically efficient". Therefore, controversies in complexity theory about the value of polynomiality of algorithms and about how to measure encoding lengths and running times – cf. Chapter 1 – were put into focus.

For almost all presently known versions of the simplex method, there exist a few (artificial) examples for which this algorithm has exponential running time. The first examples of this type have been discovered by KLEE and MINTY (1972). Such bad examples do not exist for the ellipsoid method. But the ellipsoid method has been observed to be much slower than the simplex algorithm on the average in practical computation. In fact, BORGWARDT (1982) has shown that the expected running time of a version of the simplex method is polynomial and much better than the running time of the ellipsoid method. Although the ellipsoid method does not seem to be a breakthrough in applied linear programming, it is of value in nonlinear (in particular nondifferentiable) optimization – see for instance ECKER and KUPFERSCHMID (1983).

As mentioned, nonlinear optimization is one of the roots of the ellipsoid method. The method grew out of work in convex nondifferential optimization (relaxation, subgradient, space dilatation methods, methods of central sections) as well as of studies on computational complexity of convex programming

M. Grötschel et al., *Geometric Algorithms and Combinatorial Optimization*

problems. The history of the ellipsoid method and its antecedents has been covered extensively by BLAND, GOLDFARB and TODD (1981) and SCHRADER (1982). Briefly, the development was as follows.

Based on his earlier work, SHOR (1970a,b) described a new gradient projection algorithm with space dilatation for convex nondifferential programming. YUDIN and NEMIROVSKIĬ (1976a,b) observed that Shor's algorithm provides an answer to a problem discussed by LEVIN (1965) and – in a somewhat cryptical way – gave an outline of the ellipsoid method. The first explicit statement of the ellipsoid method, as we know it today, is due to SHOR (1977). In the language of nonlinear programming, it can be viewed as a rank-one update algorithm and is quite analogous to a variable metric quasi-Newton method – see GOFFIN (1984) for such interpretations of the ellipsoid method. This method was adapted by KHACHIYAN (1979) to state the polynomial time solvability of linear programming. The proofs appeared in KHACHIYAN (1980). Khachiyan's 1979-paper stimulated a flood of research aiming at accelerating the method and making it more stable for numerical purposes – cf. BLAND, GOLDFARB and TODD (1981) and SCHRADER (1982) for surveys. We will not go into the numerical details of these modifications. Our aim is to give more general versions of this algorithm which will enable us to show that the problems discussed in Chapter 2 are equivalent with respect to polynomial time solvability and, by applying these results, to unify various algorithmic approaches to combinatorial optimization. The applicability of the ellipsoid method to combinatorial optimization was discovered independently by KARP and PAPADIMITRIOU (1980), PADBERG and RAO (1981), and GRÖTSCHEL, LOVÁSZ and SCHRIJVER (1981).

We do not believe that the ellipsoid method will become a true competitor of the simplex algorithm for practical calculations. We do, however, believe that the ellipsoid method has fundamental theoretical power since it is an elegant tool for proving the polynomial time solvability of many geometric and combinatorial optimization problems.

YAMNITSKI and LEVIN (1982) gave an algorithm – in the spirit of the ellipsoid method and also based on the research in the Soviet Union mentioned above – in which ellipsoids are replaced by simplices. This algorithm is somewhat slower than the ellipsoid method, but it seems to have the same theoretical applicability.

Khachiyan's achievement received an attention in the nonscientific press that is – to our knowledge – unpreceded in mathematics. Newspapers and journals like The Guardian, Der Spiegel, Nieuwe Rotterdamsche Courant, Népszabadság, The Daily Yomiuri wrote about the "major breakthrough in the solution of real-world problems". The ellipsoid method even jumped on the front page of The New York Times: "A Soviet Discovery Rocks World of Mathematics" (November 7, 1979). Much of the excitement of the journalists was, however, due to exaggerations and misinterpretations – see LAWLER (1980) for an account of the treatment of the implications of the ellipsoid method in the public press.

Similar attention has recently been given to the new method of KARMARKAR (1984) for linear programming. Karmarkar's algorithm uses an approach different from the ellipsoid method and from the simplex method. Karmarkar's algorithm has a better worst-case running time than the ellipsoid method, and it seems that this method runs as fast or even faster than the simplex algorithm in practice.

But Karmarkar's algorithm requires – like the simplex method – the complete knowledge of the constraint system for the linear programmming problem. And thus – as far as we can see – it cannot be used to derive the consequences to be discussed in this book.

The unexpected theoretical and practical developments in linear programming in the recent years have prompted a revival of research in this area. The nonlinear approach to linear programming – using techniques like the Newton method and related descent procedures – receives particular attention. A number of further polynomial time methods for the solution of linear programming problems have been suggested – see for instance IRI and IMAI (1986), DE GHELLINCK and VIAL (1986), BETKE and GRITZMANN (1986), and SONNEVEND (1986) – and are under investigation with respect to their theoretical and practical behaviour. It is conceivable that careful implementations of these methods and, possibly, combinations of these methods with the simplex algorithm will lead to good codes for linear programming problems. Such codes may become serious competitors for the simplex codes that presently dominate the "LP-market". A thorough computational and theoretical study of such prospects is the recent paper GILL, MURRAY, SAUNDERS, TOMLIN and WRIGHT (1986), where variants of Karmarkar's algorithm are discussed in a general framework of projected Newton barrier methods and where these are compared with several versions of the simplex method with respect to practical efficiency for various classes of LP-problems.

3.1 Geometric Background and an Informal Description

The purpose of this section is to explain the geometric idea behind the ellipsoid method, to give an informal description of it, to demonstrate some proof techniques, and to discuss various modifications. We begin by summarizing well-known geometric facts about ellipsoids. Then we describe the ellipsoid method for the special case of finding a point in a polytope that is explicitly given by linear inequalities and known to be empty or full-dimensional. We also present proofs of a few basic lemmas, and finally, computational aspects of the ellipsoid method are discussed, in particular questions of "rounding" and quicker "shrinking". Proofs of more general results than described in this section can be found in Sections 3.2 and 3.3.

The problems we address are trivial for the case of one variable. So we assume in the proofs throughout Chapter 3 that $n \geq 2$.

Properties of Ellipsoids

A set $E \subseteq \mathbb{R}^n$ is an **ellipsoid** if there exist a vector $a \in \mathbb{R}^n$ and a positive definite $n \times n$-matrix A such that

$$E = E(A, a) := \{x \in \mathbb{R}^n \mid (x-a)^T A^{-1}(x-a) \leq 1\}. \tag{3.1.1}$$

(It will become clear soon that using A^{-1} here, which is also a positive definite matrix, is more convenient than using A.) Employing the ellipsoidal norm $\| \ \|_A$

defined in Section 0.1 we can write equivalently

$$E(A,a) = \{x \in \mathbb{R}^n \mid \|x-a\|_A \leq 1\}, \tag{3.1.2}$$

that is, the ellipsoid $E(A,a)$ is the unit ball around a in the vector space $\mathbb{R}^n$ endowed with the norm $\| \ \|_A$. So in particular, the unit ball $S(0,1)$ around zero (in the Euclidean norm) is the ellipsoid $E(I,0)$. Note that E determines A and a uniquely. The vector a is called the **center** of E, and we say that $E(A,a)$ is the **ellipsoid associated with** A **and** a.

For every positive definite matrix A there exists a unique positive definite matrix, denoted by $A^{1/2}$, such that $A = A^{1/2}A^{1/2}$. It follows by a simple calculation that

$$E(A,a) = A^{1/2}S(0,1) + a. \tag{3.1.3}$$

Thus every ellipsoid is the image of the unit ball under a bijective affine transformation.

There are some interesting connections between geometric properties of the ellipsoid $E = E(A,a)$ and algebraic properties of the matrix A which we want to point out here. Recall from (0.1.3) that all eigenvalues of A are positive reals. The diameter of E is the length of a longest axis and is equal to $2\sqrt{\Lambda}$, where Λ is the largest eigenvalue of A. The longest axes of E correspond to the eigenvectors belonging to Λ. The width of E is the length of a shortest axis which is $2\sqrt{\lambda}$, where λ is the smallest eigenvalue of A. These observations imply that the ball $S(a,\sqrt{\lambda})$ is the largest ball contained in $E(A,a)$ and that $S(a,\sqrt{\Lambda})$ is the smallest ball containing $E(A,a)$. In fact, this is the geometric contents of inequality (0.1.9). Moreover, the axes of symmetry of E correspond to the eigenvectors of A.

Figure 3.1 shows an ellipsoid graphically. It is the ellipsoid $E(A,0) \subseteq \mathbb{R}^2$ with $A = \operatorname{diag}((16,4)^T)$. The eigenvalues of A are $\Lambda = 16$ and $\lambda = 4$ with corresponding eigenvectors $e_1 = (1,0)^T$ and $e_2 = (0,1)^T$. So the diameter of $E(A,0)$ is $2\sqrt{\Lambda} = 8$, while the width is $2\sqrt{\lambda} = 4$.

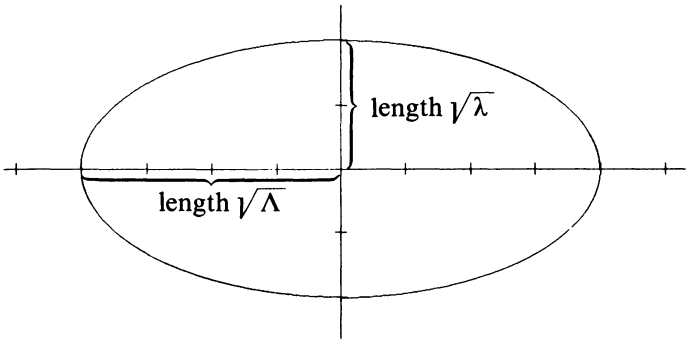


Figure 3.1

The volume of an ellipsoid $E = E(A,a)$, denoted by $\operatorname{vol}(E)$, depends only on the determinant of A and on the dimension of the space. More exactly, we have

$$\operatorname{vol}(E(A,a)) = \sqrt{\det A} \cdot V_n, \tag{3.1.4}$$

where V_n is the volume of the unit ball $S(0,1)$ in $\mathbb{R}^n$. It is well known that

$$V_n = \frac{\pi^{n/2}}{\Gamma(n/2+1)} \sim \frac{1}{\sqrt{\pi n}}\left(\frac{2e\pi}{n}\right)^{n/2}, \tag{3.1.5}$$

where

$$\Gamma(x) := \int_0^\infty e^{-t}t^{x-1}\,dt, \quad \text{for } x > 0$$

is the **gamma-function**. The gamma-function satisfies

$$\Gamma(n) = (n-1)! \quad \text{for all } n \in \mathbb{N}.$$

We will frequently need bounds on V_n. It turns out that for our purposes it will be enough to use the very rough estimates

$$n^{-n} \leq V_n \leq 2^n, \tag{3.1.6}$$

which are derived from the facts that $S(0,1)$ contains $\{x \in \mathbb{R}^n \mid 0 \leq x_i \leq 1/n, i = 1, \ldots, n\}$ and that $S(0,1)$ is contained in $\{x \in \mathbb{R}^n \mid \|x\|_\infty \leq 1\}$.

If $x \longmapsto Dx + d$ is a bijective affine transformation T then $\mathrm{vol}(T(E(A,a))) = \det D\sqrt{\det A} \cdot V_n$. This in particular shows that

$$\frac{\mathrm{vol}(E(A,a))}{\mathrm{vol}(E(B,b))} = \frac{\mathrm{vol}(T(E(A,a)))}{\mathrm{vol}(T(E(B,b)))},$$

that is, the quotient of the volumes of two ellipsoids is invariant under bijective affine transformations.

It will be necessary for us to optimize linear objective functions over ellipsoids. This is easy and can be derived from the obvious fact that, for $c \neq 0$, $\max c^T x$, $x \in S(a,1)$ is achieved at the vector $a + c/\|c\|$. Namely, suppose that $E(A,a) \subseteq \mathbb{R}^n$ is an ellipsoid and let $c \in \mathbb{R}^n \setminus \{0\}$. Set $Q := A^{1/2}$. Recall from (3.1.3) that $Q^{-1}E(A,a) = S(0,1) + Q^{-1}a = S(Q^{-1}a, 1)$ and thus

$$\begin{aligned}
\max\{c^T x \mid x \in E(A,a)\} &= \max\{c^T QQ^{-1}x \mid Q^{-1}x \in Q^{-1}E(A,a)\} \\
&= \max\{c^T Qy \mid y \in S(Q^{-1}a, 1)\} \\
&= c^T Q\frac{1}{\|Qc\|}Qc + c^T QQ^{-1}a \\
&= c^T \frac{1}{\sqrt{c^T Ac}}Ac + c^T a \\
&= c^T a + \sqrt{c^T Ac}.
\end{aligned}$$

By setting

$$\begin{aligned}
b &:= \frac{1}{\sqrt{c^T Ac}}Ac, \\
z_{\max} &:= a + b, \\
z_{\min} &:= a - b,
\end{aligned} \tag{3.1.7}$$

we therefore obtain

$$(3.1.8)\quad c^T z_{\max} = \max\{c^T x \mid x \in E(A,a)\} = c^T a + \sqrt{c^T A c} = c^T a + \|c\|_{A^{-1}},$$
$$c^T z_{\min} = \min\{c^T x \mid x \in E(A,a)\} = c^T a - \sqrt{c^T A c} = c^T a - \|c\|_{A^{-1}}.$$

This means that $z_{\max}$ maximizes and $z_{\min}$ minimizes $c^T x$ over $E(A,a)$. The line connecting $z_{\max}$ and $z_{\min}$ goes through the center a of $E(A,a)$ and has direction b. In Figure 3.2 we show the ellipsoid $E(A,a)$ with $A = \operatorname{diag}((16,4)^T)$, $a^T = (1,0.5)$ and objective function $c^T = (-2,-3)$. From (3.1.7) we obtain $b^T = (-3.2,-1.2)$, $z_{\max}^T = (1,0.5) + (-3.2,-1.2) = (-2.2,-0.7)$ and $z_{\min}^T = (1,0.5) - (-3.2,-1.2) = (4.2,1.7)$.

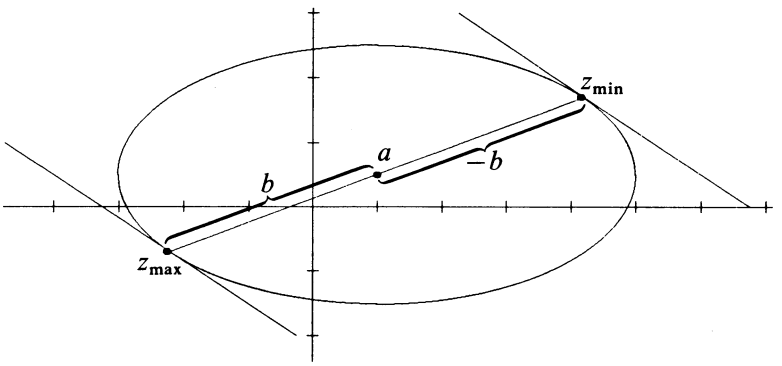


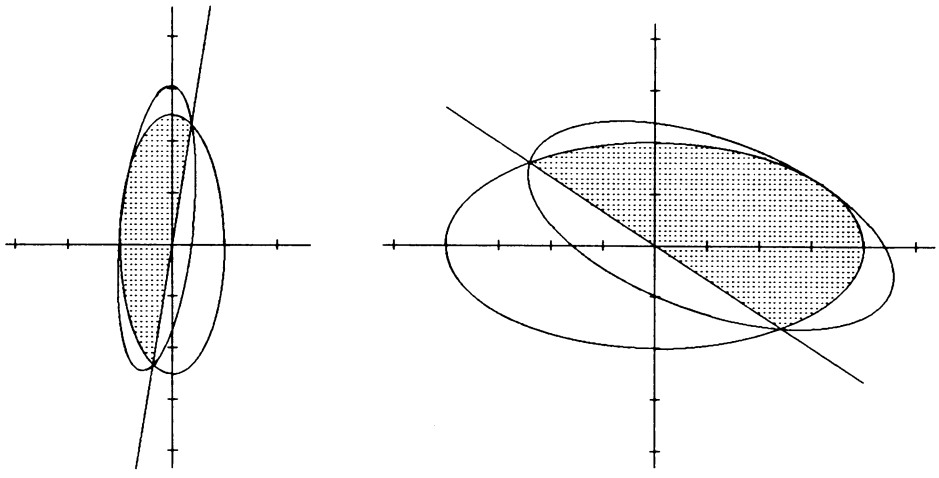
Figure 3.2

It is a well-known fact that every convex body is contained in a unique ellipsoid of minimal volume and contains a unique ellipsoid of maximal volume. Apparently, these two results have been discovered independently by several mathematicians – see for instance DANZER, GRÜNBAUM and KLEE (1963, p. 139). In particular, these authors attribute the first result to K. Löwner. JOHN (1948) proved the following more general theorem, the "moreover" part of which will be of interest later.

(3.1.9) Theorem. *For every convex body $K \subseteq \mathbb{R}^n$ there exists a unique ellipsoid E of minimal volume containing K. Moreover, K contains the ellipsoid obtained from E by shrinking it from its center by a factor of n.* □

Let us call the minimum volume ellipsoid containing a convex body K the **Löwner-John ellipsoid** of K. In formulas, the second part of Theorem (3.1.9) states that, if $E(A,a)$ is the Löwner-John ellipsoid of K, then K contains the ellipsoid $E(n^{-2}A,a)$.

For a regular simplex S, the Löwner-John ellipsoid is a ball $E(R^2 I,a)$ with an appropriate radius R around the center of gravity a of S. The concentrical ball $E(n^{-2}R^2 I,a)$ is the largest ellipsoid contained in S. This shows that the parameter n in Theorem (3.1.9) is best possible.



(a) (b)

Figure 3.3

In general, the Löwner-John ellipsoid of a convex body K is hard to compute. We will, however, show in Section 4.6 that, under certain assumptions on K, good approximations of it can be computed in polynomial time. In the ellipsoid method and its variants, Löwner-John ellipsoids of certain ellipsoidal sections are used; and for these Löwner-John ellipsoids, explicit formulas are known. We will describe some of them.

Suppose $E(A,a)$ is an ellipsoid and $c \in \mathbb{R}^n \setminus \{0\}$. Set

$$E'(A,a,c) := E(A,a) \cap \{x \in \mathbb{R}^n \mid c^T x \leq c^T a\}. \tag{3.1.10}$$

So $E'(A,a,c)$ is one half of the ellipsoid $E(A,a)$ obtained by cutting $E(A,a)$ through the center a using the hyperplane $\{x \in \mathbb{R}^n \mid c^T x = c^T a\}$. The Löwner-John ellipsoid of $E'(A,a,c)$ is the ellipsoid $E(A',a')$ given by the following formulas:

$$a' := a - \frac{1}{n+1} b, \tag{3.1.11}$$

$$A' := \frac{n^2}{n^2-1}\left(A - \frac{2}{n+1} bb^T\right), \tag{3.1.12}$$

where b is the vector defined in (3.1.7). In Figure 3.3 we show the Löwner-John ellipsoids $E(A',a')$ of two halfellipsoids $E'(A,a,c)$, where in (a), $A = \mathrm{diag}((1,25/4)^T)$, $a = 0$, $c = (25,-4)$ and in (b), $A = \mathrm{diag}((16,4)^T)$, $a = 0$, $c = (-2,-3)^T$. The halfellipsoids $E'(A,a,c)$ are the dotted regions.

Note that the center a' of the Löwner-John ellipsoid $E(A',a')$ of $E'(A,a,c)$ lies on the line through the vectors $z_{\max}$ and $z_{\min}$ – see (3.1.7). More exactly, one can

get from a to a' by making a step from a towards $z_{\min}$ of length $\frac{1}{n+1}\|z_{\min} - a\|$. Moreover, the boundary of $E(A', a')$ touches $E'(A, a, c)$ in the point $z_{\min}$ and in the set $\{x \mid \|x - a\|_A = 1\} \cap \{x \mid c^T x = c^T a\}$. In $\mathbb{R}^2$ the last set consists of two points only – see Figure 3.3 – while in $\mathbb{R}^3$ this is an ellipse.

We will see later that the algorithm described by Khachiyan makes use of the Löwner-John ellipsoid of $E'(A, a, c)$. There are modifications of this method using Löwner-John ellipsoids of other ellipsoidal sections. Here we will describe those that we use later – see BLAND, GOLDFARB and TODD (1981) for further details. So suppose $E(A, a)$ is an ellipsoid and $c \in \mathbb{R}^n \setminus \{0\}$, $\gamma \in \mathbb{R}$. It follows from (3.1.8) that the hyperplane

$$H := \{x \in \mathbb{R}^n \mid c^T x = \gamma\}$$

has a nonempty intersection with $E(A, a)$ if and only if $c^T z_{\min} \leq \gamma \leq c^T z_{\max}$, that is, if and only if

$$|c^T a - \gamma| \leq \sqrt{c^T A c}.$$

For notational convenience, let us set

$$\alpha := \frac{c^T a - \gamma}{\sqrt{c^T A c}}. \tag{3.1.13}$$

Then H intersects $E(A, a)$ if and only if

$$-1 \leq \alpha \leq 1. \tag{3.1.14}$$

The number α can be interpreted as the signed distance of the center a from the boundary of the halfspace $\{x \mid c^T x \leq \gamma\}$ in the space $\mathbb{R}^n$ endowed with the norm $\| \|_{A^{-1}}$. (The distance is nonpositive if a is contained in the halfspace.)

We want to cut $E(A, a)$ into two pieces using H and to compute the Löwner-John ellipsoid of the piece contained in $\{x \mid c^T x \leq \gamma\}$. For

$$E'(A, a, c, \gamma) := E(A, a) \cap \{x \in \mathbb{R}^n \mid c^T x \leq \gamma\}, \tag{3.1.15}$$

the Löwner-John ellipsoid $E(A', a')$ can be determined as follows.

If $-1 \leq \alpha \leq -1/n$ *then* $E(A', a') = E(A, a)$.

If $-1/n \leq \alpha < 1$ *then* $E(A', a')$ *is given by*

$$a' := a - \frac{1 + n\alpha}{n+1} b, \tag{3.1.16}$$

$$A' := \frac{n^2}{n^2 - 1}(1 - \alpha^2)\left(A - \frac{2(1 + n\alpha)}{(n+1)(1+\alpha)} bb^T\right), \tag{3.1.17}$$

where b is the vector defined in (3.1.7). Note that if $\gamma = c^T a$ then $E'(A, a, c, \gamma) = E'(A, a, c)$ and formulas (3.1.16) and (3.1.17) reduce to formulas (3.1.11) and

(3.1.12). So in formulas (3.1.11) and (3.1.12) we compute the Löwner-John ellipsoid of the section $E(A,a,c)$ of $E(A,a)$ that is obtained by cutting with H through the center a. We call this a **central cut**. If $0 < \alpha < 1$ then $E'(A,a,c,\gamma)$ is strictly contained in $E'(A,a,c)$. This means that we cut off a larger piece of $E(A,a)$, and therefore $c^T x \leq \gamma$ is called a **deep cut**. If $-1/n < \alpha < 0$ then we leave more of $E(A,a)$ than the "half" $E'(A,a,c)$, and we call $c^T x \leq \gamma$ a **shallow cut**; but in this case the Löwner-John ellipsoid of $E'(A,a,c,\gamma)$ is still strictly smaller than $E(A,a)$, in the sense that it has a smaller volume. (We shall see later that a volume reduction argument will prove the polynomial time termination of the ellipsoid method.)

It is also possible to compute explicit formulas for the Löwner-John ellipsoid of $E(A,a) \cap \{x \mid \gamma' \leq c^T x \leq \gamma\}$, i. e., for an ellipsoidal section determined by **parallel cuts**. However, the formulas are quite horrible – see BLAND, GOLDFARB and TODD (1981). We will need a special case of this, namely the Löwner-John ellipsoid of an ellipsoidal section determined by **centrally symmetric parallel cuts**, i. e., of

$$E''(A,a,c,\gamma) := E(A,a) \cap \{x \in \mathbb{R}^n \mid c^T a - \gamma \leq c^T x \leq c^T a + \gamma\}, \tag{3.1.18}$$

where $\gamma > 0$. Similarly as in (3.1.13), let $\alpha = -\gamma/\sqrt{c^T A c}$. It turns out that the Löwner-John ellipsoid of $E''(A,a,c,\gamma)$ is the ellipsoid $E(A',a')$ defined as follows.

If $\alpha < -1/\sqrt{n}$ *then* $E(A',a') = E(A,a)$.

If $-1/\sqrt{n} \leq \alpha < 0$ *then* $E(A',a')$ *is given by*

$$a' := a, \tag{3.1.19}$$

$$A' := \frac{n}{n-1}(1-\alpha^2)\left(A - \frac{1-n\alpha^2}{1-\alpha^2} bb^T\right). \tag{3.1.20}$$

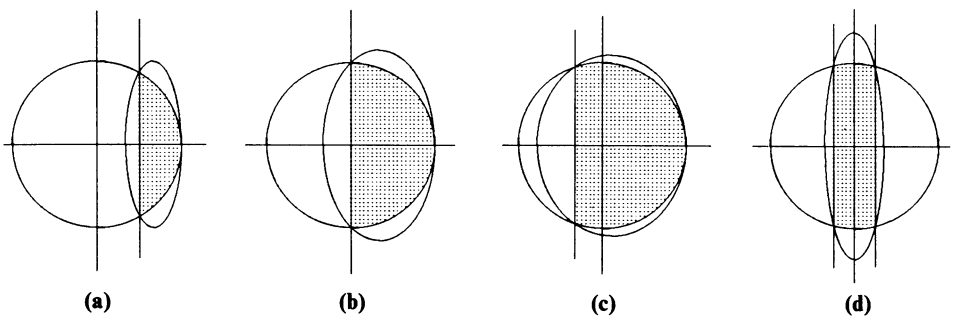


Figure 3.4a–d

Figure 3.4 shows the four types of cuts used in various versions of the ellipsoid method. In all cases we assume that $E(A,a)$ is the unit ball $S(0,1)$. Let $c :=$

$(-1,0)^T$ be the vector used to cut through $E(A,a)$. Picture (b) shows the Löwner-John ellipsoid of the dotted area $E'(A,a,c) = S(0,1) \cap \{x \mid x_1 \geq 0\}$, a central cut. Picture (a) shows the Löwner-John ellipsoid of the dotted set $E'(A,a,c,-1/2) = S(0,1) \cap \{x \mid x_1 \geq 1/2\}$, a deep cut. Picture (c) shows the Löwner-John ellipsoid of $E'(A,a,c,1/3) = S(0,1) \cap \{x \mid x_1 \geq -1/3\}$, a shallow cut; while the Löwner-John ellipsoid of $E''(A,a,c,1/4) = S(0,1) \cap \{x \mid -1/4 \leq x_1 \leq 1/4\}$ is displayed in picture (d). This set is determined by a centrally symmetric parallel cut.

Description of the Basic Ellipsoid Method

To show the basic idea behind the ellipsoid method, we describe it now as a method to solve the strong nonemptiness problem for explicitly given polytopes that are either empty or full-dimensional. The input for our algorithm is a **system of inequalities** $c_i^T x \leq \gamma_i$, $i = 1, \ldots, m$, with n variables in **integral coefficients**. We would like to determine whether

$$P := \{x \in \mathbb{R}^n \mid c_i^T x \leq \gamma_i,\ i = 1, \ldots, m\} = \{x \mid Cx \leq d\} \tag{3.1.21}$$

is empty or not, and if it is nonempty, we would like to find a point in P. In order to get a correct answer the input must be accompanied by the following guarantees:

(3.1.22) P is bounded.

(3.1.23) If P is nonempty, then P is full-dimensional.

It will turn out later that the certificates (3.1.22) and (3.1.23) are not necessary. The (appropriately modified) method also works for possibly unbounded polyhedra that are not full-dimensional. Moreover, we can handle polyhedra defined by inequalities that are provided by a separation oracle, that is, the inequalities need not be given in advance. To treat all these additional possibilities now would only obscure the lines of thought. Thus, in order to explain the ideas underlying the ellipsoid method, we restrict ourselves to the special case described above.

Recall the well-known method of catching a lion in the Sahara. It works as follows. Fence in the Sahara, and split it into two parts; check which part does not contain the lion, fence the other part in, and continue. After a finite number of steps we will have caught the lion – if there was any – because the fenced-in zone will be so small that the lion cannot move anymore. Or we realize that the fenced-in zone is so small that it cannot contain any lion, i. e., there was no lion at all. In order to illustrate the ellipsoid method by this old hunter's story we have to describe what our Sahara is, how we split it into two parts, how we fence these in, and when we can declare the lion caught or nonexistent.

For the Sahara we choose a ball around the origin, say $S(0,R)$, that contains our polytope P, the lion. If the system of inequalities (3.1.21) contains explicit upper and lower bounds on the variables, say $l_i \leq x_i \leq u_i$, $i = 1, \ldots, n$ then a radius R with $P \subseteq S(0,R)$ is easily found. Take for instance

$$R := \sqrt{\textstyle\sum_{i=1}^n \max\{u_i^2, l_i^2\}}. \tag{3.1.24}$$

If bounds on the variables are not given explicitly we can use the information that C and d are integral and that P is a polytope to compute such a radius R, namely we have:

(3.1.25) Lemma. *$P \subseteq S(0,R)$, where $R := \sqrt{n}\, 2^{\langle C,d\rangle - n^2}$.* □

Recall that $\langle C,d\rangle$ denotes the encoding length of C and d – see Section 1.3. (Since we do not want to interrupt the flow of thought, the proofs of all lemmas stated in this subsection are postponed to the next subsection.)

Now we have the starting point for the ellipsoid method. We know that P is in $S(0,R)$ with R given by (3.1.24) or (3.1.25). This ball around the origin will be our first ellipsoid $E(A_0,a_0)$ (which is clearly given by setting $A_0 := R^2 I$ and $a_0 = 0$).

Let us now describe the k-th step, $k \geq 0$, of the procedure. By construction, the current ellipsoid

$$E_k := E(A_k,a_k) \tag{3.1.26}$$

contains P. The ellipsoid E_k has one distinguished point, its center a_k, and we have it explicitly at hand. So we can substitute a_k into the inequality system (3.1.21) and check whether all inequalities are satisfied. If this is the case, we can stop, having found a feasible solution. If the center is not feasible, then at least one inequality of the system (3.1.21) must be violated, let us say $c^T x \leq \gamma$. So we have $c^T a_k > \gamma$. The hyperplane $\{x \mid c^T x = c^T a_k\}$ through the center a_k of E_k cuts E_k into two "halves", and we know from the construction that the polytope P is contained in the half

$$E'(A_k,a_k,c) = \{x \in E(A_k,a_k) \mid c^T x \leq c^T a_k\}. \tag{3.1.27}$$

Therefore we choose, as the next ellipsoid E_{k+1} in our sequence, the Löwner-John ellipsoid of $E'(A_k,a_k,c)$, which is given by formulas (3.1.11) and (3.1.12). And we continue this way by successively including P into smaller and smaller ellipsoids.

The question to be answered now is: When can we stop? Clearly, if we find a point in P, we terminate. But how long do we have to continue the iterations if no feasible point is obtained? The stopping criterion comes from a volume argument. Namely, we know the initial ellipsoid $S(0,R)$ and therefore we can estimate its volume, e. g., through (3.1.6). In each step k we construct a new ellipsoid E_{k+1} whose volume is strictly smaller than that of E_k. More precisely, one can show:

(3.1.28) Lemma.

$$\frac{\mathrm{vol}(E_{k+1})}{\mathrm{vol}(E_k)} = \left(\left(\frac{n}{n+1}\right)^{n+1}\left(\frac{n}{n-1}\right)^{n-1}\right)^{1/2} < e^{-1/(2n)} < 1.$$

□

By our guarantees (3.1.22) and (3.1.23) we are sure that our polytope P has a positive volume, unless it is empty. One can use the integrality of the data and the fact that P is a polytope to prove

(3.1.29) Lemma. *If the polytope P is full-dimensional, then*

$$\text{vol}(P) \geq 2^{-(n+1)\langle C\rangle + n^3}.$$

□

Now we can finish our analysis. We can estimate the volume of the initial ellipsoid from above, and the volume of P, if P is nonempty, from below; and we know the shrinking rate of the volumes. Therefore, we iterate until the present ellipsoid has a volume that is smaller than the lower bound (3.1.29) on the volume of P. If we have reached this situation, in step N say, we can stop since we know that

$$\begin{aligned} P &\subseteq E_N, \\ \text{vol}(E_N) &< \text{vol}(P). \end{aligned} \tag{3.1.30}$$

This is clearly impossible, and we can conclude that P is empty. The number N of iterations that have to be done in the worst case can be estimated as follows. If we choose

$$N := 2n((2n+1)\langle C\rangle + n\langle d\rangle - n^3) \tag{3.1.31}$$

then it is easy to see that $\text{vol}(E_N) < 2^{-(n+1)\langle C\rangle + n^3}$ (for an elaboration, see Lemma (3.1.36)). Combining this with (3.1.29) we see that (3.1.30) holds for this N. Our description of the ellipsoid method (except for implementational details) is complete and we can summarize the procedure.

(3.1.32) The Basic Ellipsoid Method (for the strong nonemptiness problem for full-dimensional or empty polytopes).

Input: *An $m \times n$-inequality system $Cx \leq d$ with integral coefficients. We assume that the data are accompanied by a guarantee that $P = \{x \in \mathbb{R}^n \mid Cx \leq d\}$ is bounded and either empty or full-dimensional.*

Initialization: *Set*

(a) $k := 0$,
(b) $N := 2n((2n+1)\langle C\rangle + n\langle d\rangle - n^3)$,
(c) $A_0 := R^2 I$ with $R := \sqrt{n}\, 2^{\langle C,d\rangle - n^2}$ (or use any valid smaller R as, for example, given in (3.1.24)),
$a_0 := 0$ (so $E_0 := E(A_0, a_0)$ is the initial ellipsoid).

General Step:

(d) *If $k = N$, STOP!* (Declare P empty.)
(e) *If $a_k \in P$, STOP!* (A feasible solution is found.)
(f) *If $a_k \notin P$, then choose an inequality, say $c^T x \leq \gamma$, of the system $Cx \leq d$ that is violated by a_k.*

Set

(g) $$b := \frac{1}{\sqrt{c^T A_k c}} A_k c \qquad \text{(see (3.1.7))},$$

(h) $$a_{k+1} := a_k - \frac{1}{n+1} b \qquad \text{(see (3.1.11))},$$

(i) $$A_{k+1} := \frac{n^2}{n^2-1}\left(A_k - \frac{2}{n+1} bb^T\right) \qquad \text{(see (3.1.12))},$$

and go to (d). □

We call algorithm (3.1.32) the **basic** ellipsoid method since it contains all the fundamental ideas of the procedure. To make it a polynomial time algorithm or to make it more efficient certain technical details have to be added. For instance, we have to specify how the vector a_{k+1} in (h) is to be calculated. Our formula on the right hand side of (h) leads to irrational numbers in general since a square root is computed in the formula for b in (g). Nevertheless, the lemmas stated before prove that the basic ellipsoid method works – provided that the single steps of algorithm (3.1.32) can be implemented correctly (and efficiently). We will discuss this issue in the subsection after the next one.

Proofs of Some Lemmas

We now give the proofs of the lemmas stated above. We begin with a slight generalization of Lemma (3.1.25).

(3.1.33) Lemma. *If $P = \{x \in \mathbb{R}^n \mid Cx \leq d\}$ is a polyhedron and C, d are integral, then all vertices of P are contained in the ball around 0 with radius $R := \sqrt{n}\, 2^{\langle C,d\rangle - n^2}$. In particular, if P is a polytope then $P \subseteq S(0,R)$.*

Proof. In order to show that the vertices of P are in $S(0,R)$ with the R given above, we estimate the largest (in absolute value) component of a vertex of P. Clearly, if we can find a number $t \geq 0$ such that no vertex of P has a component which (in absolute value) is larger than t, then the vertices of P are contained in $\{x \mid -t \leq x_i \leq t, i = 1, \ldots, n\}$, and thus in $S(0, \sqrt{n}\, t)$. From polyhedral theory we know that for each vertex v of P there is a nonsingular subsystem of $Cx \leq d$ containing n inequalities, say $Bx \leq b$, such that v is the unique solution of $Bx = b$. By Cramer's rule, each component v_i of v is given by

$$v_i = \frac{\det B_i}{\det B}$$

where we obtain B_i from B by replacing the i-th column of B by the vector b. Since B is integral and nonsingular, we have that $|\det B| \geq 1$, and so $|v_i| \leq |\det B_i|$. By (1.3.3) (c)

$$|\det B_i| \leq 2^{\langle B_i\rangle - n^2} - 1.$$

And thus, since $\langle B_i \rangle \leq \langle C, d \rangle$,

$$|\det B_i| \leq 2^{\langle C,d \rangle - n^2}.$$

Therefore, setting $R := \sqrt{n}\, 2^{\langle C,d \rangle - n^2}$ we can conclude that all vertices of P are contained in $S(0,R)$. □

The reader is invited to find better estimates for R, for instance by using the Hadamard inequality (0.1.28) directly and not only via (1.3.3).

Next we prove (3.1.28) and estimate how much the volume of the Löwner-John ellipsoid of the halfellipsoid $E'(A_k, a_k, c)$ shrinks compared with the volume of $E(A_k, a_k)$.

(3.1.34) Lemma. *Let $E \subseteq \mathbb{R}^n$ be an ellipsoid and E' be the ellipsoid obtained from E using formulas* (3.1.7), (3.1.11), (3.1.12) *for some vector $c \in \mathbb{R}^n \setminus \{0\}$. Then*

$$\frac{\text{vol}(E')}{\text{vol}(E)} = \left(\left(\frac{n}{n+1}\right)^{n+1}\left(\frac{n}{n-1}\right)^{n-1}\right)^{1/2} < e^{-1/(2n)} < 1.$$

Proof. To estimate the volume quotient, let us assume first, that the initial ellipsoid is $F := E(I,0)$ i. e., the unit ball around zero, and that the update vector c used to compute b in (3.1.7) is the vector $(-1,0,\ldots,0)^T$. In this case we obtain from (3.1.7), (3.1.11), and (3.1.12):

$$\begin{aligned} b &= (-1,0,\ldots,0)^T, \\ a' &= a - \frac{1}{n+1}b = \left(\frac{1}{n+1},0,\ldots,0\right)^T, \\ A' &= \frac{n^2}{n^2-1}\left(I - \frac{2}{n+1}(-1,0,\ldots,0)^T(-1,0,\ldots,0)\right) \\ &= \text{diag}\left(\left(\frac{n^2}{(n+1)^2}, \frac{n^2}{n^2-1}, \ldots, \frac{n^2}{n^2-1}\right)^T\right). \end{aligned}$$

From this and (3.1.4) we conclude for the volumes of F and $F' := E(A', a')$:

$$\begin{aligned} \frac{\text{vol}(F')}{\text{vol}(F)} &= \frac{\sqrt{\det A'}\, V_n}{\sqrt{\det A}\, V_n} = \sqrt{\det A'} = \left(\frac{n^{2n}}{(n+1)^n(n-1)^n}\,\frac{n-1}{n+1}\right)^{1/2} \\ &= \left(\left(\frac{n}{n+1}\right)^{n+1}\left(\frac{n}{n-1}\right)^{n-1}\right)^{1/2}. \end{aligned}$$

Hence, by taking the natural logarithm ln, we obtain:

$$\begin{aligned} \frac{\text{vol}(F')}{\text{vol}(F)} < e^{-1/(2n)} &\iff e^{1/n} < \left(\frac{n+1}{n}\right)^{n+1}\left(\frac{n-1}{n}\right)^{n-1} \\ &\iff \frac{1}{n} < (n+1)\ln\left(1+\frac{1}{n}\right) + (n-1)\ln\left(1-\frac{1}{n}\right). \end{aligned}$$

Using the power series expansion $\ln x = \sum_{k=1}^{\infty}(-1)^{k+1}(x-1)^k/k$ for $0 < x \leq 2$, we obtain

$$\begin{aligned}
&(n+1)\ln\left(1+\frac{1}{n}\right)+(n-1)\ln\left(1-\frac{1}{n}\right)= \\
&\quad=\sum_{k=1}^{\infty}(-1)^{k+1}\frac{n+1}{kn^k}-\sum_{k=1}^{\infty}\frac{n-1}{kn^k} \\
&\quad=\sum_{k=1}^{\infty}(-1)^{k+1}\frac{2}{kn^k}-\sum_{k=1}^{\infty}\frac{2(n-1)}{2kn^{2k}} \\
&\quad=\sum_{k=1}^{\infty}\frac{2}{(2k-1)n^{2k-1}}-\sum_{k=1}^{\infty}\frac{2}{2kn^{2k}}-\sum_{k=1}^{\infty}\frac{1}{kn^{2k-1}}+\sum_{k=1}^{\infty}\frac{1}{kn^{2k}} \\
&\quad=\sum_{k=1}^{\infty}\frac{1}{(2k-1)k}\frac{1}{n^{2k-1}} \\
&\quad>\frac{1}{n},
\end{aligned}$$

and thus the claim is proved for our special choice of the initial ellipsoid and c.

Now let $E = E(A,a)$ be any ellipsoid and $E' = E(A',a')$ be an ellipsoid constructed from E using the formulas (3.1.7), (3.1.11), and (3.1.12) for some vector $c \in \mathbb{R}^n \setminus \{0\}$. By (3.1.3), $E = A^{1/2}E(I,0) + a = A^{1/2}F + a$. Clearly, there is an orthogonal matrix, say Q, which rotates $A^{1/2}c$ into a positive multiple of $(-1,0,\ldots,0)$, that is, there is an orthogonal matrix Q such that

$$(-1,0,\ldots,0)^T = \frac{1}{\|QA^{1/2}c\|}QA^{1/2}c.$$

Then

$$T(x) := A^{1/2}Q^Tx + a$$

is a bijective affine transformation with $T^{-1}(x) = QA^{-1/2}(x-a)$. Now

$$\begin{aligned}
T(F) &= \{T(y) \mid y^Ty \leq 1\} \\
&= \{x \mid (T^{-1}x)^T(T^{-1}x) \leq 1\} \\
&= \{x \mid (x-a)^TA^{-1/2}Q^TQA^{-1/2}(x-a) \leq 1\} \\
&= \{x \mid (x-a)^TA^{-1}(x-a) \leq 1\} \\
&= E,
\end{aligned}$$

and similarly for the ellipsoid F' defined above, we obtain $T(F') = E'$. In the subsection about properties of ellipsoids we have seen that the quotient of the volumes of two ellipsoids is invariant under bijective transformations. Thus we have

$$\frac{\mathrm{vol}(E')}{\mathrm{vol}(E)} = \frac{\mathrm{vol}(T(F'))}{\mathrm{vol}(T(F))} = \frac{\mathrm{vol}(F')}{\mathrm{vol}(F)} = \left(\left(\frac{n}{n+1}\right)^{n+1}\left(\frac{n}{n-1}\right)^{n-1}\right)^{1/2} < e^{-1/(2n)},$$

which completes the proof. □

The following lemma proves (3.1.29) and shows the interesting fact that the volume of a full-dimensional polytope cannot be "extremely small", that is, a lower bound on vol(P) can be computed in polynomial time.

(3.1.35) Lemma. *If $P = \{x \in \mathbb{R}^n \mid Cx \leq d\}$ is a full-dimensional polytope and the matrix C and the vector d are integral then*

$$\text{vol}(P) \geq 2^{-(n+1)\langle C\rangle + n^3}.$$

Proof. From polyhedral theory we know that every full-dimensional polytope P in $\mathbb{R}^n$ contains $n+1$ vertices $v_0, v_1, \ldots, v_n$, say, that are affinely independent. The convex hull of these vertices forms a simplex S that is contained in P. Thus, the volume of P is bounded from below by the volume of S. The volume of a simplex can be determined by a well-known formula, namely we have

$$\text{vol}(S) = \frac{1}{n!}\left|\det \begin{pmatrix} 1 & 1 & \ldots & 1 \\ v_0 & v_1 & \ldots & v_n \end{pmatrix}\right|.$$

Now recall that, for every vertex v_j, its i-th component is of the form $\dfrac{\det B_{ji}}{\det B_j}$ by Cramers rule, where B_j is a submatrix of C and B_{ji} a submatrix of (C, d). So we obtain

$$\left|\det \begin{pmatrix} 1 & \ldots & 1 \\ v_0 & \ldots & v_n \end{pmatrix}\right| = \left|\frac{1}{\det B_0 \cdot \ldots \cdot \det B_n}\right| \left|\det \begin{pmatrix} \det B_0 & \ldots & \det B_n \\ u_0 & \ldots & u_n \end{pmatrix}\right|$$

where $u_0, \ldots, u_n$ are integral vectors in $\mathbb{R}^n$. Since the last determinant above is a nonzero integer, and since by (1.3.3) (c), $|\det B_j| \leq 2^{\langle C\rangle - n^2}$ holds, we get

$$\left|\det \begin{pmatrix} 1 & \ldots & 1 \\ v_0 & \ldots & v_n \end{pmatrix}\right| \geq \frac{1}{|\det B_0| \cdot \ldots \cdot |\det B_n|} \geq (2^{\langle C\rangle - n^2})^{-(n+1)}.$$

Therefore, using $n! \leq 2^{n^2}$, we obtain

$$\text{vol}(P) \geq \text{vol}(S) \geq \frac{1}{n!}(2^{\langle C\rangle - n^2})^{-(n+1)} \geq 2^{-(n+1)\langle C\rangle + n^3}.$$

□

The last lemma in this sequence justifies the choice (3.1.31) of the number N of iterations of the general step in the basic ellipsoid method.

(3.1.36) Lemma. *Suppose $P = \{x \in \mathbb{R}^n \mid Cx \leq d\}$ is a full-dimensional polytope and $Cx \leq d$ is an integral inequality system. If $E_0 = E(A_0, a_0)$ with $a_0 = 0$, $A_0 = R^2 I$ and $R = \sqrt{n}\, 2^{\langle C,d\rangle - n^2}$ is the initial ellipsoid, and if the general step of the basic ellipsoid method* (3.1.32) *is applied $N := 2n((2n+1)\langle C\rangle + n\langle d\rangle - n^3)$ times then*

$$\text{vol}(E_N) < 2^{-(n+1)\langle C\rangle + n^3} \leq \text{vol}(P).$$

Proof. Since $E_0 \subseteq \{x \in \mathbb{R}^n \mid \|x\|_\infty \leq R\}$, we have

$$\text{vol}(E_0) \leq 2^n R^n = n^{n/2}\, 2^{n\langle C,d\rangle - n^3 + n} = 2^{n(\langle C,d\rangle - n^2 + 1 + \log(n)/2)} < 2^{n\langle C,d\rangle}.$$

By (3.1.34), in each step of the basic ellipsoid method the volume of the next ellipsoid shrinks at least with the factor $e^{-1/(2n)}$. Thus after N steps we get

$$\text{vol}(E_N) < e^{-N/(2n)}\, \text{vol}(E_0) < 2^{-N/(2n) + n\langle C,d\rangle} = 2^{-(n+1)\langle C\rangle + n^3} \leq \text{vol}(P),$$

where the last inequality follows from (3.1.35). The equality above is, in fact, the defining equality for N. This proves our claim. □

The reader will have noticed that most of the estimates in the proofs above are quite generous. By a somewhat more careful analysis one can improve all crucial parameters slightly to obtain a smaller upper bound N on the number of iterations of the ellipsoid method. Since our main point is that the upper bound is polynomial we have chosen a short way to achieve this. As far as we can see, however, the order of magnitude of N, which is $O(n^2\langle C,d\rangle)$, cannot be improved.

Implementation Problems and Polynomiality

To estimate the running time of the basic ellipsoid method (3.1.32) in the worst case, let us count each arithmetic operation as one elementary step of the algorithm. The initialization (a), (b), (c) can obviously be done in $O(m \cdot n)$ elementary steps. In each execution of the general step (d), ..., (i) of (3.1.32) we have to substitute a_k into the system $Cx \leq d$ which requires $O(m \cdot n)$ elementary steps, and we have to update a_k and A_k in (h), (i). Clearly (h) needs $O(n)$ elementary steps while (i) requires $O(n^2)$ elementary steps. The maximum number of times the general step of (3.1.32) is executed is bounded from above by $N = 2n((2n+1)\langle C\rangle + n\langle d\rangle - n^3) = O(n^2\langle C,d\rangle)$. Since P is a polytope we have $m \geq n$, and thus the total number of elementary steps of the basic ellipsoid method is $O(mn^3\langle C,d\rangle)$. So we have found a polynomial upper bound on the number of elementary steps of the basic ellipsoid method.

However, there are certain problems concerning the implementation of some of the elementary steps. We take a square root in step (c) to calculate R, so we may obtain an irrational number. One can take care of this by rounding R up to the next integer. But, in fact, we only need the rational number R^2 in the initialization of A_0, so we do not have to compute R at all. A more serious problem comes up in the general step. The vector b calculated in (g) is in general irrational, and so is a_{k+1}. Since irrational numbers have no finite representation in binary encoding, we are not able to calculate the center a_{k+1} exactly. Thus, in any implementation of the ellipsoid method we have to round the coefficients of the entries of the center of the next ellipsoid. This causes difficulties in our analysis. Namely, by rounding a_{k+1} to some vector $\tilde{a}_{k+1} \in \mathbb{Q}^n$, say, we will translate the ellipsoid E_{k+1} slightly to the ellipsoid $\tilde{E}_{k+1} := E(A_{k+1}, \tilde{a}_{k+1})$. Recalling that E_{k+1} is the ellipsoid of minimum volume containing the halfellipsoid $E'(A_k, a_k, c)$ –

cf. (3.1.27) – we see that $\tilde{E}_{k+1}$ does not contain $E'(A_k, a_k, c)$ any more. So it may happen that the polytope $P \subseteq E'(A_k, a_k, c)$ is not contained in $\tilde{E}_{k+1}$ either. And therefore, our central proof idea of constructing a sequence of shrinking ellipsoids each containing P is not valid any more.

There is a further issue. Observe that the new matrix A_{k+1} calculated in (i) is rational, provided A_k and c are. But it is not clear a priori that the entries of A_{k+1} do not grow too fast.

All these difficulties can be overcome with some effort. We will describe here the geometric ideas behind this. The concrete mathematical analysis of the correctness and implementability of these ideas will be given in Section 3.2.

We have seen that rounding is unavoidable in the calculation of the center a_{k+1}. We will do this as follows. We write each entry of a_{k+1} in its binary representation and cut off after p digits behind the binary point, where p is to be specified. In other words, we fix a denominator 2^p and approximate each number by a rational number with this denominator. To keep the growth of the encoding lengths of the matrices A_{k+1} under control, we apply the same rounding method to the entries of A_{k+1}. As remarked before, rounding of the entries of the center results in a translation of the ellipsoid. By rounding the entries of A_{k+1} we induce a change of the shape of the ellipsoid in addition. The two roundings may produce a new ellipsoid which does not contain P. Moreover, by rounding a positive definite matrix too roughly positive definiteness may be lost. So we also have to take care that the rounded matrix is still positive definite. It follows that we should make p large enough, subject to the condition that all numbers coming up during the execution of the algorithm are of polynomial encoding length.

We still have to find a way to keep P inside the newly calculated ellipsoid. Since we cannot move P, we have to do something with the new ellipsoid. One idea that works is to maintain the (rounded) center and to blow up the ellipsoid obtained by rounding the entries of A_{k+1} in such a way that the blow-up will compensate the translation and the change of shape of the Löwner-John ellipsoid induced by rounding. The blow-up factor must be so large that the enlarged ellipsoid contains P. Clearly, we have to blow up carefully. We know from (3.1.28) that the shrinking rate is below 1; but it is very close to 1. In order to keep polynomial time termination, we have to choose a blow-up factor that on the one hand gives a sufficient shrinking rate and on the other, guarantees that P is contained in the blown-up ellipsoid. This shows that the blow-up factor and the number p determining the precision of the arithmetic influence each other, and they have to be chosen simultaneously to achieve all the goals at the same time.

It turns out that appropriate choices of N, p, and the blow-up factor ξ can be made, namely if we choose

(3.1.37)
$$\begin{aligned} N &:= 50(n+1)^2\langle C, d\rangle, \\ p &:= 8N, \\ \xi &:= 1 + \frac{1}{4(n+1)^2}, \end{aligned}$$

we can make all the desired conclusions. These modifications turn the basic ellipsoid method (3.1.32) into an algorithm that runs in polynomial time.

From the practical point of view these considerations and modifications seem quite irrelevant. Note that the ellipsoid method requires problem-specific precision. But usually, our computers have fixed precision. Even software that allows the use of variable precision would not help, since the precision demands of the ellipsoid method in this version are so gigantic that they are hardly satisfiable in practice. Thus, in a computer implementation of the ellipsoid method with fixed precision it will be possible to conclude that one of the calculated centers is contained in P, but by stopping after N steps we cannot surely declare P empty. To prove emptiness, additional tests have to be added, based for instance on the Farkas lemma.

By looking at the formulas we have stated for Löwner-John ellipsoids of various ellipsoidal sections – see (3.1.11), (3.1.12); (3.1.16), (3.1.17); (3.1.19), (3.1.20) – one can immediately see that a speed-up of the shrinking can be obtained by using deep or other cuts. There are many possibilities to "play" with the parameters of the ellipsoid method. To describe some of them let us write the basic iteration of the ellipsoid method in the following form

$$a_{k+1} :\approx a_k - \rho \frac{1}{\sqrt{c^T A_k c}} A_k c, \tag{3.1.38}$$

$$A_{k+1} :\approx \xi \cdot \sigma \left(A_k - \tau \frac{1}{c^T A_k c} A_k c c^T A_k\right), \tag{3.1.39}$$

where "$\approx$" means cutting off after the p-th digit behind the point in the binary representation of the number on the right hand side. Following BLAND, GOLDFARB and TODD (1981) we call ρ the **step parameter** (it determines the length of the step from a_k in the direction of $-b$ to obtain a_{k+1}), σ the **dilatation parameter**, τ the **expansion parameter** and ξ the **blow-up parameter** (this is the factor used to blow up the ellipsoid to compensate for the rounding errors). The ellipsoid method in perfect arithmetic (no rounding) as stated in (3.1.32) is thus given by

$$\rho := \frac{1}{n+1}, \quad \sigma := \frac{n^2}{n^2-1}, \quad \tau := \frac{2}{n+1}, \quad \xi := 1. \tag{3.1.40}$$

It is a so-called central-cut method since it always uses cuts through the center of the current ellipsoid. The ellipsoid method with rounding and blow-up, i. e., the polynomial time version of (3.1.32) is thus defined by

$$\rho := \frac{1}{n+1}, \quad \sigma := \frac{n^2}{n^2-1}, \quad \tau := \frac{2}{n+1}, \quad \xi := 1 + \frac{1}{4(n+1)^2}. \tag{3.1.41}$$

If $c^T x \leq \gamma$ is the violated inequality found in (f) of (3.1.32), then defining $\alpha := (c^T a_k - \gamma)/\sqrt{c^T A_k c}$ as in (3.1.13) and setting

$$\rho := \frac{1+n\alpha}{n+1}, \quad \sigma := \frac{n^2(1-\alpha^2)}{n^2-1}, \quad \tau := \frac{2(1+n\alpha)}{(n+1)(1+\alpha)}, \quad \xi := 1 \tag{3.1.42}$$

we obtain the **deep-cut ellipsoid method** (in perfect arithmetic). Note that since $c^T a_k > \gamma$, the hyperplane $\{x \mid c^T x = \gamma\}$ is indeed a deep cut, and so the Löwner-John ellipsoid of $E'(A_k, a_k, c, \gamma)$ – see (3.1.15) – which is given through formulas (3.1.38), (3.1.39) (without rounding), and (3.1.42) has indeed a smaller volume than that of $E'(A_k, a_k, c)$ used in (3.1.32).

The use of deep cuts will – due to faster shrinking – in general speed up the convergence of the ellipsoid method. But one can also easily construct examples where the standard method finds a feasible solution more quickly. However, the use of deep cuts does not seem to change the order of magnitude of the number N of iterations of the general step necessary to correctly conclude that P is empty. So from a theoretical point of view, deep cuts do not improve the worst-case running time of the algorithm.

Using a number α in (3.1.42) with $-1/n < \alpha < 0$ we obtain a **shallow-cut ellipsoid method**. Although – from a practical point of view – it seems ridiculous to use a shallow-cut method, since we do not shrink as much as we can, it will turn out that this version of the ellipsoid method is of particular theoretical power. Using this method we will be able to prove results which – as far as we can see – cannot be derived from the central-cut or deep-cut ellipsoid method.

Some Examples

The following examples have been made up to illustrate the iterations of the ellipsoid method geometrically. Let us first consider the polytope $P \subseteq \mathbb{R}^2$ defined by the inequalities

$$\begin{array}{lrcl} (1) & -x_1 - x_2 & \leq & -2 \\ (2) & 3x_1 & \leq & 4 \\ (3) & -2x_1 + 2x_2 & \leq & 3. \end{array}$$

This polytope is contained in the ball of radius 7 around zero. To find a point in P, we start the basic ellipsoid method (3.1.32) with $E(A_0, a_0)$ where

$$A_0 = \begin{pmatrix} 49 & 0 \\ 0 & 49 \end{pmatrix}, \quad a_0 = \begin{pmatrix} 0 \\ 0 \end{pmatrix},$$

see Figure 3.5. The center a_0 violates inequality (1). One iteration of the algorithm yields the ellipsoid $E(A_1, a_1)$ also shown in Figure 3.5. The new center a_1 violates (2). We update and continue this way. The fifth iteration produces the ellipsoid $E(A_5, a_5)$ displayed in Figure 3.6. In the 7-th iteration the ellipsoid $E(A_7, a_7)$ shown in Figure 3.7 is found. Its center $a_7 = (1.2661, 2.3217)^T$ is contained in P. The whole sequence of ellipsoids $E(A_0, a_0), \ldots, E(A_7, a_7)$ produced in this example is shown in Figure 3.8.

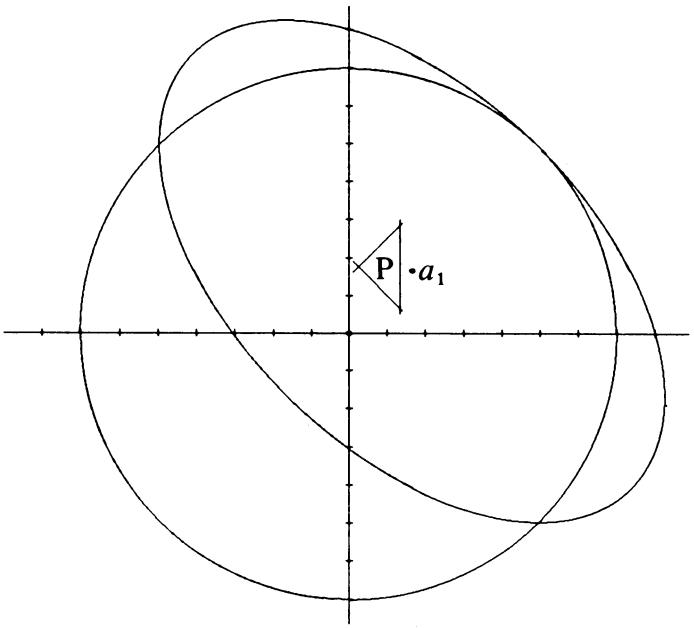


Figure 3.5

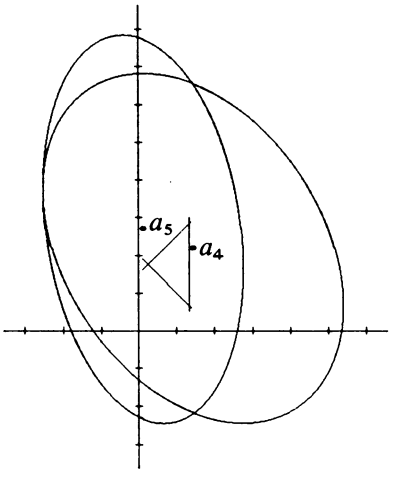


Figure 3.6

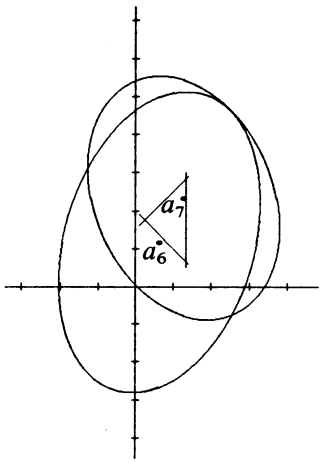


Figure 3.7

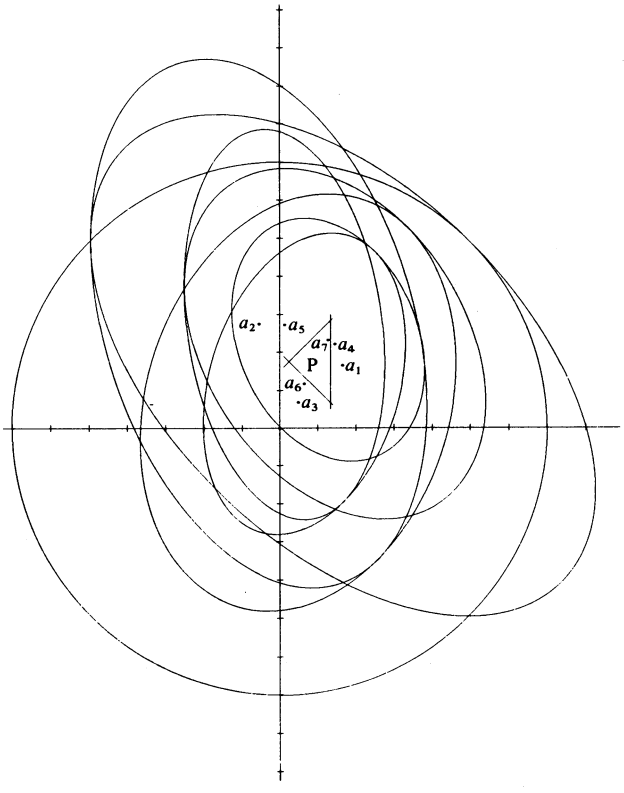
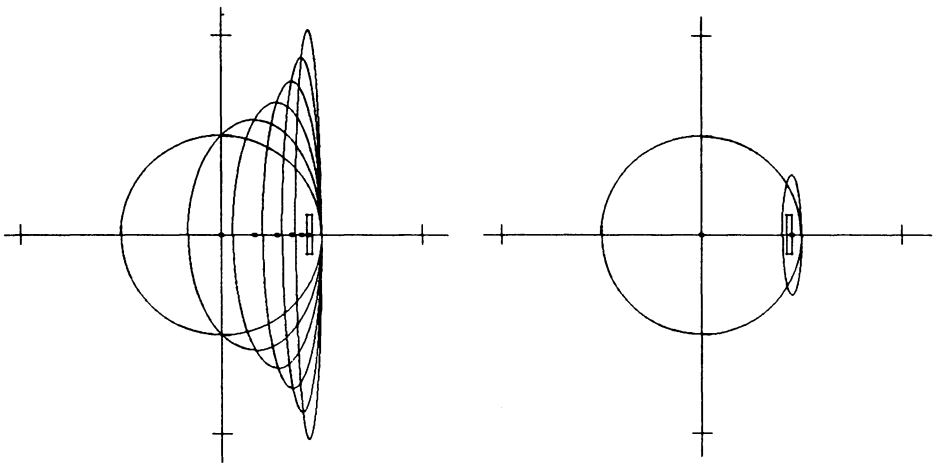


Figure 3.8

The pictures of the iterations of the ellipsoid method show clearly that in an update the ellipsoid is squeezed in the direction of c while it is expanded in the direction orthogonal to c. If many updates are done with one and the same vector c the ellipsoids become "needles". To demonstrate this effect, let us consider the polytope $P \subseteq \mathbb{R}^2$ defined by

$$\frac{17}{20} \leq x_1 \leq \frac{18}{20}$$
$$-\frac{1}{5} \leq x_2 \leq \frac{1}{5}.$$

Starting the basic ellipsoid method with $E(A_0, a_0) = S(0, 1)$, we make five iterations until the center $a_5 = (211/243, 0)^T$ of the fifth ellipsoid $E(A_5, a_5)$ is contained in P. The six ellipsoids $E(A_0, a_0), \ldots, E(A_5, a_5)$ of this sequence are displayed in Figure 3.9 (a). If the ellipsoid method receives "flat" polytopes as its input, it is likely that the algorithm produces extremely flat ellipsoids, say of several kilometers length and a few millimeters width. This inevitably leads to numerical difficulties in practice. In such cases, some variants of the ellipsoid method frequently perform better empirically. For instance, the deep-cut ellipsoid method finds a feasible point of the polytope P defined above in one iteration. This iteration is shown in Figure 3.9 (b). But there is no general result quantifying this improvement in performance.



(a) (b)

Figure 3.9

*3.2 The Central-Cut Ellipsoid Method

We shall now give a description and analysis of the version of the basic ellipsoid method (3.1.32) where arithmetic operations are performed in finite precision and the errors induced by rounding are compensated for by blowing up the

"rounded" Löwner-John ellipsoid. This section provides proofs of the claims about this method made in the previous section. In particular, the polynomial running time is established. Moreover, we do not restrict ourselves to the case of an explicitly given polytope, but we treat the general case of a circumscribed closed convex set given by a certain separation oracle. Our main result of this section can be stated as follows.

(3.2.1) Theorem. *There exists an oracle-polynomial time algorithm, called the* **central-cut ellipsoid method**, *that solves the following problem:*

Input: *A rational number $\varepsilon > 0$ and a circumscribed closed convex set $(K; n, R)$ given by an oracle SEP_K that, for any $y \in \mathbb{Q}^n$ and any rational number $\delta > 0$, either asserts that $y \in S(K, \delta)$ or finds a vector $c \in \mathbb{Q}^n$ with $\|c\|_\infty = 1$ such that $c^T x \leq c^T y + \delta$ for every $x \in K$.*

Output: *One of the following:*

(i) *a vector $a \in S(K, \varepsilon)$,*

(ii) *a positive definite matrix $A \in \mathbb{Q}^{n \times n}$ and a point $a \in \mathbb{Q}^n$ such that $K \subseteq E(A, a)$ and $\mathrm{vol}(E(A, a)) \leq \varepsilon$.*

Whenever we speak of the ellipsoid method without referring to a special version we will mean the central-cut ellipsoid method, to be described in the proof of Theorem (3.2.1). This method specifies a sequence of ellipsoids $E_0, E_1, \ldots, E_N$ (i. e., a sequence of positive definite matrices $A_0, A_1, \ldots, A_N$, and a sequence of centers $a_0, a_1, \ldots, a_N$, with $E_k = E(A_k, a_k)$, $k = 0, \ldots, N$), that contain the given set K, such that either at least one of the centers a_k satisfies $a_k \in S(K, \varepsilon)$ (so alternative (i) of (3.2.1) is achieved), or the last ellipsoid E_N has volume at most ε.

Proof of Theorem (3.2.1). The proof will be given in several steps. We first describe the method, then prove its correctness assuming that several lemmas hold. Finally the truth of the lemmas will be established.

So let numbers n, R, ε and an oracle SEP_K be given as required in the theorem. Without loss of generality $\varepsilon < 1$.

I. For the precise description of the algorithm we need the following parameters:

$$N := \lceil 5n|\log \varepsilon| + 5n^2|\log(2R)|\rceil, \tag{3.2.2}$$

$$p := 8N, \tag{3.2.3}$$

$$\delta := 2^{-p}. \tag{3.2.4}$$

The integer N is the maximum number of iterations of the central-cut ellipsoid method, the rational number δ is the error we allow the oracle SEP_K to make, and the integer p is the precision parameter for representation of numbers. We assume throughout the algorithm that all numbers occurring are represented in binary form and are rounded to p digits behind the point. Clearly, for every rational number given by a numerator and denominator, this binary approximation can be easily computed.

We initialize the procedure by setting:

$$a_0 := 0, \tag{3.2.5}$$
$$A_0 := R^2 I,$$

so that $E_0 = S(0, R)$ and hence $K \subseteq E_0$.

Assume a_k, A_k are found for some $k \geq 0$. If $k = N$, then the ellipsoid $E_N = E(A_N, a_N)$ has the property required in (ii) of (3.2.1) (we shall prove this later), and we stop.

If $k < N$, we call the oracle SEP_K with $y = a_k$ and the error parameter δ defined in (3.2.4).

If SEP_K concludes that $a_k \in S(K, \delta)$, then by the choice of δ, $a_k \in S(K, \varepsilon)$, and we stop having achieved (i) of (3.2.1).

If SEP_K gives a vector $c \in \mathbb{Q}^n$ with $\|c\|_\infty = 1$ and $c^T x \leq c^T a_k + \delta$ for all $x \in K$, then we do the following computations:

$$a_{k+1} :\approx a_k - \frac{1}{n+1} \frac{A_k c}{\sqrt{c^T A_k c}}, \tag{3.2.6}$$

$$A_{k+1} :\approx \frac{2n^2+3}{2n^2} \left(A_k - \frac{2}{n+1} \frac{A_k c c^T A_k}{c^T A_k c} \right), \tag{3.2.7}$$

where the sign "$\approx$" in (3.2.6), (3.2.7) means that the left hand side is obtained by cutting the binary expansions of the numbers on right hand side after p digits behind the binary point. This finishes the description of the central-cut ellipsoid method.

II. To prove the correctness of the algorithm, we establish the following facts. (Recall that for a vector x, $\|x\|$ denotes the Euclidean norm, and for a matrix A, $\|A\|$ denotes the spectral norm (0.1.16).)

(3.2.8) Lemma. *The matrices $A_0, A_1, \ldots$ are positive definite. Moreover,*

$$\|a_k\| \leq R2^k, \quad \|A_k\| \leq R^2 2^k, \quad \text{and} \quad \|A_k^{-1}\| \leq R^{-2} 4^k.$$

(3.2.9) Lemma. *$K \subseteq E_k$ for $k = 0, 1, \ldots$*

(3.2.10) Lemma. *$\mathrm{vol}(E_{k+1}) / \mathrm{vol}(E_k) \leq e^{-1/(5n)}$ for $k = 0, 1, \ldots$*

We shall prove these lemmas in III, IV, and V below.

It follows from Lemma (3.2.8) that all the formulas on the right hand sides of (3.2.6), (3.2.7) are meaningful (no division by 0) and that the intermediate numbers (entries of a_k and A_k) do not grow too large, i. e., have polynomial encoding lengths. This shows that all the arithmetic operations can be carried out in polynomial time.

If the algorithm stops with $a_k \in S(K, \varepsilon)$, then of course we have nothing to prove. If it stops with $k = N$, then by Lemma (3.2.9), E_N indeed contains K. Moreover, by Lemma (3.2.10) the volume of E_N satisfies

$$\mathrm{vol}(E_N) \leq e^{-N/(5n)} \mathrm{vol}(E_0).$$

E_0 is the ball around zero with radius R, so E_0 is contained in the hypercube $Q = \{x \in \mathbb{R}^n \mid -R \leq x_i \leq R,\ i = 1, \ldots, n\}$. From the very rough estimate $\text{vol}(E_0) \leq \text{vol}(Q) = (2R)^n$ we get

$$\text{vol}(E_N) \leq e^{-N/(5n)}(2R)^n < 2^{-N/(5n)}(2R)^n \leq \varepsilon. \tag{3.2.11}$$

The last inequality in (3.2.11) in fact is the inequality from which the value of N is derived. So the truth of this inequality directly follows from the choice of N. This finishes the proof of Theorem (3.2.1) subject to the correctness of (3.2.8), (3.2.9), (3.2.10), which we show now. □

III. **Proof of Lemma** (3.2.8). We prove the lemma by induction on k. Since $a_0 = 0$ and since the largest eigenvalue of A_0 is R^2, all the statements of (3.2.8) are clearly true for $k = 0$. Assume that they are true for $k \geq 0$. Let a_{k+1}^*, A_{k+1}^* be the right hand sides of (3.2.6) and (3.2.7) without rounding, i. e.,

$$a_{k+1}^* := a_k - \frac{1}{n+1}\frac{A_k c}{\sqrt{c^T A_k c}}, \tag{3.2.12}$$

$$A_{k+1}^* := \frac{2n^2+3}{2n^2}\left(A_k - \frac{2}{n+1}\frac{A_k c c^T A_k}{c^T A_k c}\right). \tag{3.2.13}$$

Then note first that

$$(A_{k+1}^*)^{-1} = \frac{2n^2}{2n^2+3}\left(A_k^{-1} + \frac{2}{n-1}\frac{cc^T}{c^T A_k c}\right), \tag{3.2.14}$$

which is easy to verify by computation. Thus, $(A_{k+1}^*)^{-1}$ is the sum of a positive definite and a positive semidefinite matrix, and so it is positive definite. Hence also A_{k+1}^* is positive definite.

Equation (0.1.17) immediately implies that for positive semidefinite matrices A and B, $\|A\| \leq \|A+B\|$ holds. Using this, the fact that A_{k+1}^* is positive definite, and the induction hypothesis, we have:

$$\begin{aligned}\|A_{k+1}^*\| &= \frac{2n^2+3}{2n^2}\left\|A_k - \frac{2}{n+1}\frac{A_k c c^T A_k}{c^T A_k c}\right\| \\ &\leq \frac{2n^2+3}{2n^2}\|A_k\| \leq \frac{11}{8}R^2 2^k.\end{aligned} \tag{3.2.15}$$

Further, since each entry of A_{k+1} differs from the corresponding entry of A_{k+1}^* by at most 2^{-p} (due to our rounding procedure) we have from (0.1.23)

$$\|A_{k+1} - A_{k+1}^*\| \leq \|A_{k+1} - A_{k+1}^*\|_{\max} \leq n2^{-p}. \tag{3.2.16}$$

So (3.2.15), (3.2.16), and the choice of p give

$$\|A_{k+1}\| \leq \|A_{k+1} - A_{k+1}^*\| + \|A_{k+1}^*\| \leq n2^{-p} + \frac{11}{8}R^2 2^k \leq R^2 2^{k+1}. \tag{3.2.17}$$

This proves the second claim of Lemma (3.2.8).

Moreover, setting $Q := A_k^{1/2}$, using formula (3.2.12) and the definition of the spectral norm (0.1.16) we obtain

$$\|a_{k+1}^* - a_k\| = \frac{1}{n+1}\frac{\|A_k c\|}{\sqrt{c^T A_k c}} = \frac{1}{n+1}\sqrt{\frac{c^T A_k^2 c}{c^T A_k c}} = \frac{1}{n+1}\sqrt{\frac{(c^T Q^T)A_k(Qc)}{c^T Q^T Q c}} \tag{3.2.18}$$
$$\leq \frac{1}{n+1}\sqrt{\|A_k\|} \leq \frac{1}{n+1}R2^{k-1}.$$

Our rounding procedure and (0.1.7) give

$$\|a_{k+1} - a_{k+1}^*\| \leq \sqrt{n}\|a_{k+1} - a_{k+1}^*\|_\infty \leq \sqrt{n}2^{-p}; \tag{3.2.19}$$

and therefore, by the induction hypothesis, the choice of p, and the inequalities (3.2.18) and (3.2.19) derived above we get

$$\begin{aligned}\|a_{k+1}\| &\leq \|a_{k+1} - a_{k+1}^*\| + \|a_{k+1}^* - a_k\| + \|a_k\| \\ &\leq \sqrt{n}2^{-p} + \frac{1}{n+1}R2^{k-1} + R2^k \\ &\leq R2^{k+1}.\end{aligned} \tag{3.2.20}$$

This proves the first claim of (3.2.8). Finally, we observe that

$$\begin{aligned}\|(A_{k+1}^*)^{-1}\| &\leq \frac{2n^2}{2n^2+3}\left(\|A_k^{-1}\| + \frac{2}{n-1}\frac{\|cc^T\|}{c^T A_k c}\right) \\ &\leq \frac{2n^2}{2n^2+3}\left(\|A_k^{-1}\| + \frac{2}{n-1}\|A_k^{-1}\|\right) \\ &< \frac{n+1}{n-1}\|A_k^{-1}\| \\ &\leq 3R^{-2}4^k.\end{aligned} \tag{3.2.21}$$

(The first inequality above follows from (3.2.14). To get the second inequality, note that (0.1.16) immediately yields $\|cc^T\| = c^T c$. Setting $Q = A_k^{1/2}$, by (0.1.18) we have

$$\frac{\|cc^T\|}{c^T A_k c} = \frac{c^T c}{c^T Q^2 c} \leq \|Q^{-1}\|^2 = \|A_k^{-1}\|.$$

The last inequality in (3.2.21) follows from $n \geq 2$ and our induction hypothesis.)

Let λ_0 denote the least eigenvalue of A_{k+1} and let v be a corresponding eigenvector with $\|v\| = 1$. Then by (3.2.16), (3.2.21), and (0.1.17), (0.1.23), and the choice of p:

$$\begin{aligned}\lambda_0 = v^T A_{k+1} v &= v^T A_{k+1}^* v + v^T(A_{k+1} - A_{k+1}^*)v \\ &\geq \|(A_{k+1}^*)^{-1}\|^{-1} - \|A_{k+1} - A_{k+1}^*\| \\ &> \frac{1}{3}R^2 4^{-k} - n2^{-p} \\ &\geq R^2 4^{-(k+1)}.\end{aligned} \tag{3.2.22}$$

From $\lambda_0 > 0$ we can conclude that A_{k+1} is positive definite. Moreover, by (0.1.18) and (3.2.22),

$$\|A_{k+1}^{-1}\| = \lambda_0^{-1} \leq R^{-2}4^{k+1}.$$

This proves the third assertion of (3.2.8). □

IV. **Proof of Lemma** (3.2.9). By induction on k. The claim holds by construction for $k = 0$. Suppose the claim is true for k, and let a_{k+1}^* and A_{k+1}^* be the vector resp. matrix defined in (3.2.12) and (3.2.13) in the proof of the previous lemma. Take any $x \in K$. We have to prove that

$$(x - a_{k+1})^T A_{k+1}^{-1}(x - a_{k+1}) \leq 1. \tag{3.2.23}$$

We shall estimate the left-hand side of (3.2.23) in several steps. Using (3.2.14) we get

$$\begin{aligned}(3.2.24)\quad (x - a_{k+1}^*)^T (A_{k+1}^*)^{-1}(x - a_{k+1}^*) &= \\ &= \frac{2n^2}{2n^2+3}\Big(x - a_k + \frac{1}{n+1}\frac{A_k c}{\sqrt{c^T A_k c}}\Big)^T \\ &\qquad \Big(A_k^{-1} + \frac{2}{n-1}\frac{cc^T}{c^T A_k c}\Big)\Big(x - a_k + \frac{1}{n+1}\frac{A_k c}{\sqrt{c^T A_k c}}\Big) \\ &= \frac{2n^2}{2n^2+3}\Big((x-a_k)^T A_k^{-1}(x-a_k) + \\ &\qquad + \frac{1}{n^2-1} + \frac{2}{n-1}\frac{c^T(x-a_k)}{\sqrt{c^T A_k c}} + \frac{2}{n-1}\frac{(c^T(x-a_k))^2}{c^T A_k c}\Big).\end{aligned}$$

By induction hypothesis, we know that $K \subseteq E_k$. So x belongs to E_k, and thus the first term in the last formula of (3.2.24) above is at most 1. Setting

$$t := \frac{c^T(x - a_k)}{\sqrt{c^T A_k c}}, \tag{3.2.25}$$

we therefore obtain

$$(x - a_{k+1}^*)^T (A_{k+1}^*)^{-1}(x - a_{k+1}^*) \leq \frac{2n^2}{2n^2+3}\Big(\frac{n^2}{n^2-1} + \frac{2}{n-1}t(t+1)\Big). \tag{3.2.26}$$

To estimate $t(t+1)$, we proceed as follows. Using the fact that A_k can be written as QQ, where $Q = A_k^{1/2}$, we can bound t from above employing the Cauchy-Schwarz inequality (0.1.26):

$$\begin{aligned}|c^T(x - a_k)| &= |c^T Q(Q^{-1}(x - a_k))| \\ &\leq \|c^T Q\| \|Q^{-1}(x - a_k)\| \\ &= \sqrt{c^T QQc}\ \sqrt{(x-a_k)^T Q^{-1}Q^{-1}(x-a_k)} \\ &= \sqrt{c^T A_k c}\ \sqrt{(x-a_k)^T A_k^{-1}(x-a_k)},\end{aligned}$$

and hence, since $x \in E_k$

$$|t| = \Big|\frac{c^T(x-a_k)}{\sqrt{c^T A_k c}}\Big| \leq \sqrt{(x-a_k)^T A_k^{-1}(x-a_k)} \leq 1. \tag{3.2.27}$$

The oracle SEP_K guarantees that $c^T(x-a_k) \le \delta$, and so

$$t = \frac{c^T(x-a_k)}{\sqrt{c^T A_k c}} \le \frac{\delta}{\|c\|\sqrt{\|A_k^{-1}\|^{-1}}} \le \delta\sqrt{\|A_k^{-1}\|} \le \delta R^{-1}2^k.$$

Using this estimate and (3.2.27), we can conclude

$$t(t+1) \le 2\delta R^{-1}2^N.$$

Substituting this upper bound for $t(t+1)$ in (3.2.26), we get

$$\begin{aligned}(3.2.28)\quad (x-a_{k+1}^*)^T(A_{k+1}^*)^{-1}(x-a_{k+1}^*) &\le \frac{2n^2}{2n^2+3}\left(\frac{n^2}{n^2-1} + 4\delta R^{-1}2^N\right)\\ &\le \frac{2n^4}{2n^4+n^2-3} + 4\delta R^{-1}2^N.\end{aligned}$$

The rest of the proof is standard error estimation:

$$\begin{aligned}\Delta &:= |(x-a_{k+1})^T A_{k+1}^{-1}(x-a_{k+1}) - (x-a_{k+1}^*)^T(A_{k+1}^*)^{-1}(x-a_{k+1}^*)|\\ &\le |(x-a_{k+1})^T A_{k+1}^{-1}(a_{k+1}^*-a_{k+1})| + |(a_{k+1}^*-a_{k+1})A_{k+1}^{-1}(x-a_{k+1}^*)|\\ &\quad + |(x-a_{k+1}^*)^T(A_{k+1}^{-1} - (A_{k+1}^*)^{-1})(x-a_{k+1}^*)|\\ &\le \|x-a_{k+1}\|\,\|A_{k+1}^{-1}\|\,\|a_{k+1}^*-a_{k+1}\| + \|a_{k+1}^*-a_{k+1}\|\,\|A_{k+1}^{-1}\|\,\|x-a_{k+1}^*\|\\ &\quad + \|x-a_{k+1}^*\|^2\|A_{k+1}^{-1}\|\,\|(A_{k+1}^*)^{-1}\|\,\|A_{k+1}^*-A_{k+1}\|.\end{aligned}$$

To continue the estimation, we observe that by Lemma (3.2.8), by the choice of p, and by the facts that $x \in S(0,R)$ and $k \le N-1$:

$$\|x-a_{k+1}\| \le \|x\| + \|a_{k+1}\| \le R + R2^{k+1} \le R2^{N+1},$$

$$\|x-a_{k+1}^*\| \le \|x-a_{k+1}\| + \|a_{k+1}-a_{k+1}^*\| \le R + R2^{k+1} + \sqrt{n}\,2^{-p} \le R2^{N+1}.$$

So again by (3.2.8) and by (3.2.21), we conclude:

$$\begin{aligned}(3.2.29)\quad \Delta &\le (R2^{N+1})(R^{-2}4^N)(\sqrt{n}\,2^{-p}) + (\sqrt{n}\,2^{-p})(R^{-2}4^N)(R2^{N+1})\\ &\quad + (R^2 2^{2N+2})(R^{-2}4^N)(R^{-2}4^N)(n2^{-p})\\ &\le nR^{-1}2^{3N+2-p} + nR^{-2}2^{6N+2-p}.\end{aligned}$$

Now we can put the estimates (3.2.28) and (3.2.29) together to get (3.2.23):

$$\begin{aligned}(x-a_{k+1})^T\, &A_{k+1}^{-1}(x-a_{k+1}) \le\\ &\le |(x-a_{k+1})^T A_{k+1}^{-1}(x-a_{k+1}) - (x-a_{k+1}^*)(A_{k+1}^*)^{-1}(x-a_{k+1}^*)|\\ &\quad + (x-a_{k+1}^*)^T(A_{k+1}^*)^{-1}(x-a_{k+1}^*)\\ &\le \frac{2n^4}{2n^4+n^2-3} + 4\delta R^{-1}2^N + nR^{-1}2^{3N+2-p} + nR^{-2}2^{6N+2-p}\\ &\le 1,\end{aligned}$$

where the last inequality follows from the choice of p. □

V. **Proof of Lemma** (3.2.10). As mentioned in (3.1.4), the volume of an ellipsoid $E(A,a) \subseteq \mathbb{R}^n$ is known to be $\sqrt{\det(A)}\ \mathrm{vol}(S(0,1))$. Hence we obtain

$$\frac{\mathrm{vol}(E_{k+1})}{\mathrm{vol}(E_k)} = \sqrt{\frac{\det(A_{k+1})}{\det(A_k)}} = \sqrt{\frac{\det(A^*_{k+1})}{\det(A_k)}}\ \sqrt{\frac{\det(A_{k+1})}{\det(A^*_{k+1})}}, \tag{3.2.30}$$

where A^*_{k+1} is the matrix defined in (3.2.13). To estimate the first factor on the right hand side of (3.2.30), write $A_k = QQ$ where $Q = A_k^{1/2}$ and use the definition of A^*_{k+1}:

$$\frac{\det(A^*_{k+1})}{\det(A_k)} = \det(Q^{-1}A^*_{k+1}Q^{-1}) = \Big(\frac{2n^2+3}{2n^2}\Big)^n \det\Big(I - \frac{2}{n+1}\frac{Qcc^TQ}{c^TQQc}\Big).$$

Since Qcc^TQ/c^TQQc has rank one and trace one, the matrix in the last determinant has eigenvalues $1, \ldots, 1, 1-2/(n+1)$. So

$$\frac{\det(A^*_{k+1})}{\det(A_k)} = \Big(\frac{2n^2+3}{2n^2}\Big)^n \frac{n-1}{n+1} \le e^{3/(2n)}e^{-2/n} = e^{-1/(2n)}. \tag{3.2.31}$$

To obtain the last estimate we have used the well-known facts that $1 + x \le e^x$ for all $x \in \mathbb{R}$ and $(1+\frac{2}{n-1})^n > e^2$, for $n \ge 2$ – cf. PÓLYA and SZEGÖ (1978), I. 172.

To estimate the second factor of (3.2.30) write it as follows (and recall inequality (3.2.21) and the fact that $\det B \le \|B\|^n$):

$$\begin{aligned}
\frac{\det A_{k+1}}{\det A^*_{k+1}} &= \det(I + (A^*_{k+1})^{-1}(A_{k+1} - A^*_{k+1})) \\
&\le \|I + (A^*_{k+1})^{-1}(A_{k+1} - A^*_{k+1})\|^n \\
&\le (\|I\| + \|(A^*_{k+1})^{-1}\|\,\|A_{k+1} - A^*_{k+1}\|)^n \\
&\le (1 + (R^{-2}4^{k+1})(n2^{-p}))^n \\
&\le e^{n^2 2^{2N-p}R^{-2}} \\
&\le e^{1/(10n)},
\end{aligned} \tag{3.2.32}$$

where the last inequality follows from the choice of N and p. Thus, from the estimates (3.2.31) and (3.2.32) we get

$$\frac{\mathrm{vol}(E_{k+1})}{\mathrm{vol}(E_k)} \le \sqrt{\frac{\det A^*_{k+1}}{\det A_k}}\ \sqrt{\frac{\det A_{k+1}}{\det A^*_{k+1}}} \le e^{-1/(4n)+1/(20n)} = e^{-1/(5n)}.$$

This finishes the proof of Lemma (3.2.10). □

Thus the proof of Theorem (3.2.1) is complete.

(3.2.33) Remark. Suppose that, instead of the oracle SEP_K in the input of Theorem (3.2.1), we have an oracle SEP_{K,K_1} (for some fixed subset $K_1 \subseteq K$), which is weaker than SEP_K in the following sense: for any $y \in \mathbb{Q}^n$ and any rational $\delta > 0$, SEP_{K,K_1} either asserts that $y \in S(K,\delta)$ or finds a vector $c \in \mathbb{Q}^n$ with $\|c\|_\infty = 1$ such that $c^T x \leq c^T y + \delta$ for every $x \in K_1$. Then we obtain the same conclusion as in (3.2.1) except that we can only guarantee that K_1 is contained in $E(A,a)$. □

(3.2.34) Remark. For technical reasons, the "separation" oracle used in Theorem (3.2.1) is slightly stronger than a weak separation oracle. However, for well-bounded convex bodies (where we know an inner radius for the convex set), any weak separation oracle can be turned into one necessary for Theorem (3.2.1). This is immediate from the following lemma. □

(3.2.35) Lemma. *Let $(K;n,R,r)$ be a well-bounded convex body, $c \in \mathbb{R}^n$ with $\|c\|_\infty = 1$, and $\gamma,\delta \in \mathbb{R}$ with $r > \delta > 0$. Suppose that $c^T x \leq \gamma$ is valid for $S(K,-\delta)$. Then*

$$c^T x \leq \gamma + \frac{2R\delta}{r}\sqrt{n} \quad \textit{for all } x \in K.$$

Proof. Let x_0 be a point in K maximizing $c^T x$ over K. By definition (2.1.16), there is a point $a_0 \in \mathbb{R}^n$ with $S(a_0,r) \subseteq K$. Consider the function $f(x) := \frac{\delta}{r}x + (1-\frac{\delta}{r})x_0$. Set $a := f(a_0)$, then $f(S(a_0,r)) = S(a,\delta)$, and moreover, $S(a,\delta)$ is contained in $\mathrm{conv}(\{x_0\} \cup S(a_0,r))$. Since $x_0 \in K$, and $S(a_0,r) \subseteq K$, $S(a,\delta)$ is contained in K by convexity. Hence $a \in S(K,-\delta)$, and so we know that $c^T a \leq \gamma$. Now note that $x_0 = a + \frac{\delta}{r}(x_0 - a_0)$. Thus in order to estimate $c^T x_0$, we have to find a bound on $c^T(x_0 - a_0)$. Since $x_0, a_0 \in K$ and $K \subseteq S(0,R)$, we can conclude that $\|x_0 - a_0\| \leq 2R$, and since $\|c\|_\infty = 1$, we know that $\|c\| \leq \sqrt{n}$. Putting this together and using the Cauchy-Schwarz inequality (0.1.26) we get:

$$c^T x_0 \leq c^T(a + \frac{\delta}{r}(x_0 - a_0)) \leq \gamma + \frac{\delta}{r}c^T(x_0 - a_0) \leq \gamma + \frac{\delta}{r}\|c\|\|x_0 - a_0\| \leq \gamma + \frac{\delta}{r}\sqrt{n}\,2R,$$

and the claim is proved. □

*3.3 The Shallow-Cut Ellipsoid Method

We have already mentioned before that the central-cut ellipsoid method as described in the previous section does not make full use of the geometric idea behind it. It has been observed by many authors that, for instance, using deep cuts can speed up the method (see SHOR and GERSHOVICH (1979), and BLAND, GOLDFARB and TODD (1981) for a survey). A deeper idea due to YUDIN and NEMIROVSKIĬ (1976b) is the use of shallow cuts. These provide slower (though still polynomial time) termination but work for substantially weaker separation oracles. This method will allow us to derive (among other results) the polynomial

time equivalence of the weak membership (2.1.14) and weak separation problems (2.1.13) for centered convex bodies.

To formulate this method precisely, we have to define a **shallow separation oracle** for a convex set K. Before giving an exact definition, we describe the geometric idea on which the shallow-cut method is based. In the central-cut ellipsoid method we stop as soon as we have found a point almost in the convex set K. Now we want to find a point deep in K – even more, we are looking for an ellipsoid $E(A,a)$ containing K such that the concentrical ellipsoid $E((n+1)^{-2}A,a)$ is contained in K. The method stops as soon as such an ellipsoid $E(A,a)$ is found. If an ellipsoid $E(A,a)$ does not have this property, then $E((n+1)^{-2}A,a) \setminus K$ is nonempty, and we look for a halfspace $c^T x \leq \gamma$ which contains K but does not completely contain $E((n+1)^{-2}A,a)$. Such a halfspace will be called a **shallow cut** since it may contain the center a of $E(A,a)$ in its interior – see Figure 3.10.

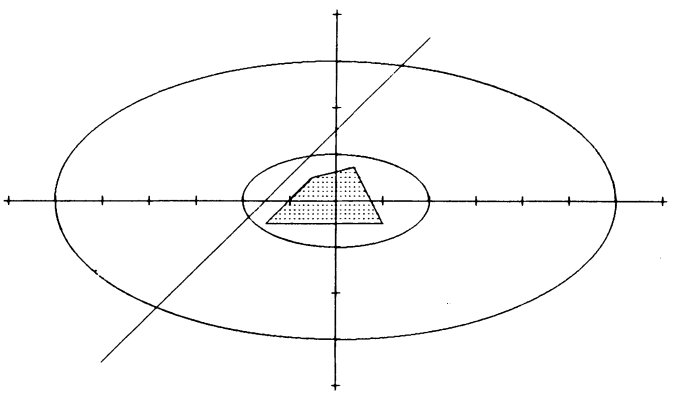


Figure 3.10

The method proceeds by determining the minimum volume ellipsoid containing

$$E(A,a) \cap \{x \mid c^T x \leq c^T a + \frac{1}{n+1}\sqrt{c^T A c}\} \tag{3.3.1}$$

and continues this way. Of course, since irrational numbers may come up, it will be necessary to round, and therefore the Löwner-John-ellipsoid has to be blown up a little bit as in the central-cut ellipsoid method.

Note that, by (3.1.8), the right hand side $c^T a+(n+1)^{-1}\sqrt{c^T A c} = c^T a+(n+1)^{-1}$ $\|c\|_{A^{-1}}$ in (3.3.1) is the maximum value the linear function $c^T x$ assumes on the ellipsoid $E((n+1)^{-2}A,a)$. So the halfspace $\{x \in \mathbb{R}^n \mid c^T x \leq c^T a+(n+1)^{-1}\sqrt{c^T A c}\}$ contains this ellipsoid and supports it at the point $a+((n+1)\sqrt{c^T A c})^{-1}Ac$.

(3.3.2) Definition. *A* **shallow separation oracle for a convex set** $K \subseteq \mathbb{R}^n$ *is an oracle whose input is an ellipsoid $E(A,a)$ described by a positive definite matrix $A \in \mathbb{Q}^{n\times n}$ and a vector $a \in \mathbb{Q}^n$. A shallow separation oracle for K can write one of the following two possible answers on its output tape:*

(i) *a vector* $c \in \mathbb{Q}^n$, $c \neq 0$, *so that the halfspace* $H := \{x \in \mathbb{R}^n \mid c^T x \leq c^T a + (n+1)^{-1}\sqrt{c^T A c}\}$ *contains* $K \cap E(A,a)$ *(a vector* c *with this property is called a* **shallow cut** *for* K *and* $E(A,a)$*),*

(ii) *the assertion that* $E(A,a)$ *is* **tough**.

□

At least two remarks are necessary to explain this definition. In answer (i), the inequality $c^T x \leq \gamma$ with $\gamma = c^T a + \frac{1}{n+1}\sqrt{c^T A c}$ defining the halfspace H containing $K \cap E(A,a)$ has an irrational right hand side in general. But note that this right hand side γ is not written on the output tape. The oracle only confirms that H contains $K \cap E(A,a)$. In answer (ii) we have used the yet undefined word "tough". Loosely speaking, the word "tough" stands for "cutting is impossible". "Toughness" is a parameter left open, and in every instance of a shallow separation oracle the particular meaning of "tough" has to be specified. For instance, in the example described above a tough ellipsoid would be an ellipsoid $E(A,a)$ such that $E((n+1)^{-2}A, a)$ is contained in K. But there are other meaningful and interesting definitions of toughness possible.

We assume, as usual, that with each shallow separation oracle a polynomial function Φ is associated such that for every input to the oracle of encoding length at most L the encoding length of its output is at most $\Phi(L)$.

The aim of this section is to prove the following.

(3.3.3) Theorem. *There exists an oracle-polynomial time algorithm, called the* **shallow-cut ellipsoid method**, *that, for any rational number* $\varepsilon > 0$ *and for any circumscribed closed convex set* $(K; n, R)$ *given by a shallow separation oracle, finds a positive definite matrix* $A \in \mathbb{Q}^{n \times n}$ *and a point* $a \in \mathbb{Q}^n$ *such that one of the following holds:*

(i) $E(A,a)$ *has been declared tough by the oracle,*

(ii) $K \subseteq E(A,a)$ *and* $\mathrm{vol}(E(A,a)) \leq \varepsilon$.

□

Before giving a proof of a slightly more general version of this theorem, we want to illustrate it by three special cases.

(3.3.4) Example. Suppose that $K \subseteq \mathbb{R}^n$, $n \geq 2$, is a full-dimensional polytope given as the solution set of a system of linear inequalities

$$a_i^T x \leq \alpha_i, \quad i = 1, \ldots, m,$$

where $a_i \in \mathbb{Q}^n$, $\alpha_i \in \mathbb{Q}$ for $i = 1, \ldots, m$. We design a shallow separation oracle as follows. Let $E(A,a)$ be an ellipsoid given by A and a. For $i = 1, \ldots, m$, determine whether all points in the ellipsoid $E((n+1)^{-2}A, a)$ satisfy $a_i^T x \leq \alpha_i$. It follows from (3.1.8) that this can be done by checking whether $(n+1)^{-2} a_i^T A a_i \leq (\alpha_i - a_i^T a)^2$ holds. If an index i is found for which this inequality does not hold, then the oracle gives the shallow cut a_i as answer. If all inequalities of the given system are satisfied by all points in $E((n+1)^{-2}A, a)$, then the oracle declares $E(A,a)$ tough.

If L denotes the encoding length of the inequality system $a_i^T x \leq \alpha_i$, $i = 1, \ldots, m$, then we know from Lemma (3.1.33) that $K \subseteq S(0, R(K))$ with $R(K) := \sqrt{n}\, 2^{L-n^2}$ and from Lemma (3.1.35) that $\text{vol}(K) \geq \varepsilon(K) := 2^{-(n+1)L+n^3}$. So, by running the shallow-cut ellipsoid method of Theorem (3.3.3) with input $(K; n, R(K))$ and $\varepsilon = \varepsilon(K)$ and with the shallow separation oracle defined above, we will obtain an ellipsoid $E(A, a)$ containing K such that the concentrical ellipsoid $E((n+1)^{-2}A, a)$ is contained in K. □

Example (3.3.4) applies to more general situations. Namely, if for a circumscribed convex set K we have a shallow separation oracle where toughness of an ellipsoid $E(A, a)$ means that $E((n+1)^{-2}A, a) \subseteq K$, then the shallow-cut ellipsoid method solves the weak nonemptiness problem for K with the additional advantage that it gives a point deep inside K.

(3.3.5) Example. The central-cut ellipsoid method can be simulated by the shallow-cut ellipsoid method, and Theorem (3.2.1) is a consequence of Theorem (3.3.3). In fact, suppose that we have an oracle SEP_K as in (3.2.1) for a circumscribed closed convex set $(K; n, R)$ and that a positive rational number $\varepsilon > 0$ is given. Then we can design a shallow separation oracle for K as follows. Let an ellipsoid $E(A, a)$ be given by A and a. Compute a positive strict lower bound ε_1 for the square root of the least eigenvalue λ of A. Let $\delta' := \min\{\varepsilon, \varepsilon_1\}$ and $\delta := (n+1)^{-1}\delta'$. Call the oracle SEP_K for K with input $y := a$ and error parameter δ. If the oracle SEP_K asserts that $y \in S(K, \delta)$, then we declare the ellipsoid $E(A, a)$ tough. If SEP_K finds a vector $c \in \mathbb{Q}^n$ with $\|c\|_\infty = 1$ and $c^T x \leq c^T y + \delta$ for all $x \in K$, then we take the vector c as output of the shallow separation oracle. By the choice of δ, the vector c is indeed a shallow cut for K and $E(A, a)$, namely, for all $x \in K$ we have

$$
\begin{aligned}
(3.3.6) \qquad c^T x &\leq c^T a + \delta \leq c^T a + (n+1)^{-1}\varepsilon_1 \leq c^T a + (n+1)^{-1}\sqrt{\lambda} \\
&= c^T a + (n+1)^{-1}\sqrt{\lambda}\, \|c\|_\infty \leq c^T a + (n+1)^{-1}\sqrt{\lambda}\, \|c\| \\
&\leq c^T a + (n+1)^{-1}\|c\|_{A^{-1}},
\end{aligned}
$$

where the last inequality follows from (0.1.9). In this case, toughness implies that the center of the tough ellipsoid is in $S(K, \delta)$ and hence in $S(K, \varepsilon)$. □

(3.3.7) Example. We can also turn a weak separation oracle into a shallow separation oracle provided an inner radius r for K is known. This follows directly by combining Remark (3.2.33) with Example (3.3.5). □

By definition (3.3.2), a shallow cut for K and $E(A, a)$ is a vector $c \in \mathbb{Q}^n$ such that $c^T x \leq c^T a + (n+1)^{-1}\sqrt{c^T A c}$ for all $x \in K \cap E(A, a)$. The parameter $(n+1)^{-1}$ used in the right hand side of this inequality is just a convenient choice out of an interval of possible parameters with which a shallow-cut method can be defined and for which it works. For our applications, greater generality is not necessary. But we will state and prove a slightly more general theorem to show what can be done.

(3.3.8) Definition. *For any rational number β with $0 < \beta < 1/n$ a* **shallow β-separation oracle for a convex set** *$K \subseteq \mathbb{R}^n$ is an oracle which, for an input a, A, where $a \in \mathbb{Q}^n$ and A is a rational positive definite $n \times n$-matrix, writes one of the following two answers on its output tape:*

(i) *a vector $c \in \mathbb{Q}^n$, $c \neq 0$, such that the halfspace $\{x \mid c^T x \leq c^T a + \beta\sqrt{c^T A c}\}$ contains $K \cap E(A,a)$ (such a vector c is called a* **shallow β-cut for K and $E(A,a)$**),

(ii) *the assertion that $E(A,a)$ is tough.* □

Observe that the halfspace $\{x \in \mathbb{R}^n \mid c^T x \leq c^T a + \beta\sqrt{c^T A c}\}$ contains and supports the ellipsoid $E(\beta^2 A, a)$. Clearly a shallow separation oracle as defined in (3.3.2) is a shallow $\frac{1}{n+1}$-oracle as defined above.

(3.3.9) Theorem. *There exists an algorithm, called the* **shallow-β-cut ellipsoid method**, *that, for any $\beta \in \mathbb{Q}$, $0 < \beta < 1/n$, and for any circumscribed closed convex set $(K; n, R)$ given by a shallow β-separation oracle, and for any rational $\varepsilon > 0$, finds, in time oracle-polynomial in $n + \langle R\rangle + \langle\varepsilon\rangle + \lceil(1-n\beta)^{-1}\rceil$, a positive definite matrix $A \in \mathbb{Q}^{n\times n}$ and a vector $a \in \mathbb{Q}^n$ such that one of the following holds:*

(i) *$E(A,a)$ has been declared tough by the oracle;*

(ii) *$K \subseteq E(A,a)$ and $\mathrm{vol}(E(A,a)) \leq \varepsilon$.*

Note that the algorithm we are going to design is not polynomial in the encoding length $\langle\beta\rangle$ of β. But if we choose β such that the encoding length of the number $(1-n\beta)^{-1}$ is bounded by a polynomial in the encoding length $n + \langle R\rangle + \langle\varepsilon\rangle$ of the other input (e. g., if we set $\beta := (n+1)^{-1}$), then the algorithm is truely oracle-polynomial. So Theorem (3.3.3) directly follows from Theorem (3.3.9).

Proof of Theorem (3.3.9). As in the proof of Theorem (3.2.1) we are going to describe a sequence of ellipsoids $E_0, E_1, \ldots$,i. e., we are going to construct a sequence of positive definite matrices $A_0, A_1, \ldots$ and a sequence of centers $a_0, a_1, \ldots$ such that $E_k = E(A_k, a_k)$. The algorithm we describe is the **shallow-β-cut ellipsoid method**. Set

$$N := \left\lceil \frac{5n}{(1-n\beta)^2} |\log\varepsilon| + \frac{5n^2}{(1-n\beta)^2} |\log(2R)| + |\log(1-n\beta)| \right\rceil \tag{3.3.10}$$

$$p := 8N. \tag{3.3.11}$$

We initialize the procedure by setting

$$\begin{aligned} a_0 &:= 0 \\ A_0 &:= R^2 I. \end{aligned} \tag{3.3.12}$$

Assume a_k, A_k are defined for some $k \geq 0$. If $k = N$, we stop. In this case the ellipsoid E_N contains K and has volume at most ε, so alternative (ii) of (3.3.9) is achieved. If $k < N$, we call the shallow β-separation oracle with $a = a_k$ and $A = A_k$.

If the oracle concludes that $E(A, a)$ is tough, then E_k has the desired property.

If the oracle gives a shallow β-cut c, then we perform the following computations.

$$a_{k+1} :\approx a_k - \rho \frac{A_k c}{\sqrt{c^T A_k c}}, \tag{3.3.13}$$

$$A_{k+1} :\approx A'_{k+1} := \zeta \cdot \sigma\left(A_k - \tau \frac{A_k c c^T A_k}{c^T A_k c}\right), \tag{3.3.14}$$

where

$$\rho := \frac{1 - n\beta}{n+1}, \tag{3.3.15}$$

$$\sigma := \frac{n^2(1-\beta^2)}{n^2-1}, \tag{3.3.16}$$

$$\tau := \frac{2(1-n\beta)}{(n+1)(1-\beta)}, \tag{3.3.17}$$

$$\zeta := 1 + \frac{(1-n\beta)^2}{2n^2}. \tag{3.3.18}$$

Again "$\approx$" means that the left hand side is obtained by rounding the right hand side to p digits behind the point. (Note that without rounding and without blowing-up (i. e., with setting $\zeta := 1$) the update formulas above determine the Löwner-John-ellipsoid of $E'_k(A, a, c, \gamma)$ with $\gamma := c^T a_k + \beta\sqrt{c^T A_k c}$ – cf. (3.1.15), (3.1.16), (3.1.17).)

Similarly as in the case of the central-cut ellipsoid method, to establish the correctness of the algorithm we need the following lemmas.

(3.3.19) Lemma. *The matrices $A_0, A_1, \ldots$ are positive definite. Moreover,*

$$\|a_k\| \le R2^k, \ \|A_k\| \le R^2 2^k, \text{and} \|A_k^{-1}\| \le R^{-2}4^k.$$

(3.3.20) Lemma. $K \subseteq E_k$ *for* $k = 0, 1, \ldots$.

(3.3.21) Lemma. $\operatorname{vol}(E_{k+1})/\operatorname{vol}(E_k) \le e^{-(1-n\beta)^2/(5n)}$ *for* $k = 0, 1, \ldots$.

The first two of these lemmas can be proved along the same lines as Lemmas (3.2.8) and (3.2.9). We will prove Lemma (3.3.21), where the crucial condition $\beta < 1/n$ plays a role.

Proof of Lemma (3.3.21). As in the proof of Lemma (3.2.10) we write

$$\frac{\operatorname{vol}(E_{k+1})}{\operatorname{vol}(E_k)} = \sqrt{\frac{\det(A'_{k+1})}{\det(A_k)}} \sqrt{\frac{\det(A_{k+1})}{\det(A'_{k+1})}}, \tag{3.3.22}$$

where A_{k+1} is defined in (3.3.14), and we obtain for the first factor in (3.3.22)

$$\text{(3.3.23)}\quad \sqrt{\frac{\det(A_{k+1}^*)}{\det(A_k)}} = \sqrt{(\zeta\sigma)^n(1-\tau)} = \zeta^{n/2}\sigma^{(n-1)/2}\sqrt{\sigma(1-\tau)}$$
$$= \Big(1+\frac{(1-n\beta)^2}{2n^2}\Big)^{n/2}\Big(\frac{n^2(1-\beta^2)}{n^2-1}\Big)^{(n-1)/2}\frac{n(1+\beta)}{n+1}.$$

The first of the three factors in (3.3.23) can be easily estimated as follows:

$$\text{(3.3.24)}\qquad \Big(1+\frac{(1-n\beta)^2}{2n^2}\Big)^{n/2} \le e^{(1-n\beta)^2/(4n)}.$$

To derive an upper bound for the last two factors in (3.3.23) take the natural logarithm ln (and recall the power series expansion of $\ln(1+x)$ and $\ln(1-x)$):

$$\text{(3.3.25)}\quad \ln\Big(\Big(\frac{n^2(1-\beta^2)}{n^2-1}\Big)^{(n-1)/2}\cdot\frac{n(1+\beta)}{n+1}\Big) =$$
$$= \frac{n-1}{2}\Big(\ln(1-\beta^2)-\ln\Big(1-\frac{1}{n^2}\Big)\Big)+\ln(1+\beta)-\ln\Big(1+\frac{1}{n}\Big)$$
$$= \frac{n-1}{2}\Big(\sum_{k=1}^{\infty}\frac{1}{k}\Big(\frac{1}{n^{2k}}-\beta^{2k}\Big)\Big)+\sum_{k=1}^{\infty}\frac{(-1)^k}{k}\Big(\frac{1}{n^k}-\beta^k\Big)$$
$$= n\Big(\sum_{k=1}^{\infty}\frac{1}{2k}\Big(\frac{1}{n^{2k}}-\beta^{2k}\Big)\Big)-\sum_{k=1}^{\infty}\frac{1}{2k-1}\Big(\frac{1}{n^{2k-1}}-\beta^{2k-1}\Big)$$
$$= \sum_{k=1}^{\infty}\frac{-1}{2k(2k-1)n^{2k-1}}((2k-1)(n\beta)^{2k}-2k(n\beta)^{2k-1}+1)$$
$$\le \frac{-(1-n\beta)^2}{2n}$$

The last inequality follows from the observation that each term of the series on the left hand side is negative as $n\beta < 1$. Hence the first term $-(1-n\beta)^2/(2n)$ of this last sum is an upper bound for it.

Thus, from (3.3.24) and (3.3.25) we get

$$\text{(3.3.26)}\qquad \sqrt{\frac{\det(A_{k+1}^*)}{\det(A_k)}} \le e^{(1-n\beta)^2/(4n)}e^{-(1-n\beta)^2/(2n)} = e^{-(1-n\beta)^2/(4n)}$$

The second factor in (3.3.22) can be estimated just like in the proof of Lemma (3.2.10), and we obtain

$$\text{(3.3.27)}\qquad \sqrt{\frac{\det(A_{k+1})}{\det(A_{k+1}^*)}} \le e^{(1-n\beta)^2/(20n)}.$$

Combining inequalities (3.3.26) and (3.3.27) gives the desired result. This completes the proof of Lemma (3.3.21) and, by the same argument as in the proof of Theorem (3.2.1), also the proof of Theorem (3.3.9). □

As mentioned above, we will only use the shallow β-cut ellipsoid method for $\beta = \frac{1}{n+1}$ in the sequel, and that is what we call the shallow-cut ellipsoid method. Similarly, if $\beta = \frac{1}{n+1}$, a shallow β-separation oracle is called just a **shallow separation oracle**. The parameters used in the shallow-cut ellipsoid method are the following (compare with (3.3.10), ..., (3.3.18) and (3.2.2), ..., (3.2.7)):

$$
\begin{aligned}
N &: \lceil 5n(n+1)^2 |\log \varepsilon| + 5n^2(n+1)^2 |\log(2R)| + \log(n+1) \rceil, \\
p &:= 8N, \\
\rho &:= \frac{1}{(n+1)^2}, \\
\sigma &:= \frac{n^3(n+2)}{(n+1)^3(n-1)}, \\
\tau &:= \frac{2}{n(n+1)}, \\
\zeta &:= 1 + \frac{1}{2n^2(n+1)^2}.
\end{aligned}
$$

So, in particular one can see that the number N of iterations of the shallow-cut ellipsoid method is about $(n+1)^2$ times as large as the number of iterations of the central-cut ellipsoid method. This, of course, matters for practical purposes, but is of no significance if one is only interested in polynomial time solvability.