Minimum Spanning Ellipsoids

Mark J. Post

Department of Electrical Engineering and Computer Science
The Johns Hopkins University
Baltimore, Maryland 21218

Abstract

The notion of a minimum spanning ellipsoid in any dimension is explained. Basic definitions and theorems provide the ideas for an algorithm to find the minimum spanning ellipsoid of a set of points, i.e., the ellipsoid of minimum volume containing the set. The run-time of the algorithm $O(n^2)$ independent of dimension, where n is the number of points.

Introduction

The problem of finding the Minimum Spannning Ellipse of a set of planar, convex points was introduced in [Po81]. An algorithm which ran in $O(n^3)$ time was described to compute the smallest ellipse containing the n points. This algorithm was clarified and improved in [Po82, Po83], yielding a $O(n^2)$ algorithm which also could be modified for non-convex sets of points. This paper introduces the notion of minimum spanning ellipsoids, and the basic concepts necessary for computing them independent of dimension. Proofs of the various properties presented are not included in this paper.

The Structure of Ellipsoids

For our discussion, we will assume that the points come from a d-dimensional, euclidean co-ordinate space, E^d. Given an ordered pair, $(\vec{x}_0, A)$, where $\vec{x}_0$ is a point in d-space (column vector), and A is a $d \times d$ positive-definite matrix, the following theorem defines the *center form* for an ellipsoid.

Theorem 1. The set of points which satisfy the equation:

$$(\vec{x} - \vec{x}_0)^t \, A \, (\vec{x} - \vec{x}_0) = 1$$

defines an ellipsoid centered at $\vec{x}_0$.

The volume of an ellipsoid in its center form is given by the following corollary.

Corollary 1. The volume of an ellipsoid E: $(\vec{x}_0, A)$ equals $\frac{v_d}{\sqrt{\det(A)}}$, where v_d is the volume of the unit hypersphere in dimension d.

Consider the left-hand side of the equation of Theorem 1. It represents a function with parameters $(\vec{x}_0, A)$. This function is linear with respect to A and quadratic with respect to $\vec{x}_0$. An alternate form similar to Theorem 1 can be derived with the property that the left-hand side of its defining equation will be *linear* in its parameters.

Theorem 2. Let E be an ellipsoid in E^d. Then for some constant c, E is the set of points which satisfy an equation of the form:

$$\vec{x}^t A\vec{x} + \vec{b}^t \vec{x} + c = 0$$

where A is a $d \times d$ positive-definite matrix and $\vec{b}^t$ is a row-vector. We denote this *linear form* by E: $(A, \vec{b}, c)$.

Corollary 2. Let E be an ellipsoid in its linear form. Then the volume of E is equal to

$$\frac{v_d \cdot \left[-c + \frac{\vec{b}^t A^{-1} \vec{b}}{4} \right]^{\frac{d}{2}}}{\sqrt{\det(A)}}$$

This theorem concludes the discussion of the structure of ellipsoids in E^d. They provide the basic representation needed to compute minimum spanning ellipsoids. First, however, the next section defines the notion of spanning, and proves various convex properties for ellipsoids in their standard forms.

Spanning and Convexity

To start this discussion, we define what it means for an ellipsoid to span a set of points. Consider a finite set of points P, $|P| = n$, and an ellipsoid E: $(\vec{x}_0, A)$ in its center form. Let $\vec{x} \in P$.

Definition 1. If $(\vec{x} - \vec{x}_0)^t A (\vec{x} - \vec{x}_0) \leq 1$, then E is said to *span* $\vec{x}$. If the inequality is strict, then E *properly* spans $\vec{x}$.

Let E: $(A, \vec{b}, c)$ be an ellipsoid in its linear form and let P be a set of points. Consider the following notation. For a point $\vec{x} \in E^d$, we use the notation $E(\vec{x})$ to represent the value obtained by substituting $\vec{x}$ in the linear form equation of E. That is,

$$E(\vec{x}) = \vec{x}^t A\, \vec{x} + \vec{b}^t \vec{x} + c.$$

The following definition defines the spanning property for linear form ellipsoids.

Definition 2. If $E(\vec{x}) \leq 0$, then E spans $\vec{x}$. E *properly* spans $\vec{x}$ if the inequality is strick.

We now consider some convex properties of center form ellipsoids. First, let E: $(\vec{x}_0, A)$ and E': $(\vec{x}_0{}', A')$, $\vec{x}_0 \neq \vec{x}_0{}'$, be ellipsoids in their center forms. Then, we define the ellipsoid formed by taking the convex combination of E and E':

$$E_\alpha: \left[(\alpha\vec{x}_0 + \bar{\alpha}\vec{x}_0{}'), (\alpha A + \bar{\alpha} A') \right]$$

Theorem 3. Let $\vec{x}$ be spanned by E and E'. Then $\vec{x}$ is properly spanned by E_α.

This theorem has shown that spanning is a convex property of center form ellipsiods with different centers. When the two ellipsoids have different centers, this theorem has shown that E_α *properly* spans all the points, including those on E and E'. Thus, in general, being ON an ellispoid is NOT a convex property of center form ellipsoids.

Theorem 4. Volume is a convex funtion of center form ellipsoids. Thus,

$$Vol(E_\alpha) \leqslant \alpha \cdot Vol(E) + \bar{\alpha} \cdot Vol(E')$$

We have now shown that spanning and volume are convex properties of center form ellipsoids. We have also shown that being on an ellipsoid is NOT a convex property of arbitrary center form ellipsoids. We now show that the linear form has similar, but slightly different properties. Let E: $(A, \vec{b}, c)$ and E': $(A', \vec{b}', c')$ be linear form ellipsoids. Now, consider the convex combination of E and E':

$$E_\alpha = (\alpha \cdot E + \bar{\alpha} \cdot E'): \left\{(\alpha \cdot A + \bar{\alpha} \cdot A'), (\alpha \cdot \vec{b} + \bar{\alpha} \cdot \vec{b}'), (\alpha \cdot c + \bar{\alpha} \cdot c')\right\}$$

Theorem 5. Let $\vec{x}$ be spanned by E and E'. Then $\vec{x}$ is spanned by E_α.

This theorem shows that any point spanned by E and E' is also spanned by E_α. Although being ON center form ellipsoids is not convex, the following corrollary shows this property IS convex for linear form ellipsoids.

Corollary 3. If $\vec{x}$ is on E and E', then for all α, $\vec{x}$ is on E_α.

Corollary 3 illustrates the usefulness of the linear form in analyzing the set of ellipsoids which go through some common set of points. The linearity of the defining equation also allows the parameters of such an ellipsoid to be multiplied by a scalar without changing the defined ellipsoid. This property is important when analyzing the volume of linear form ellipsoids. Center form ellipsoids provided structure which made the analysis of their volumes easy. Corollary 3 shows that the linear form has some spanning properties which the center form does not. However, the volume of an ellipsoid in such a form has a complicated structure. Thus, the analysis of the volume becomes more difficult.

Volume, in general, is not a convex function of linear formed ellipsoids. However, it is a function which does have an important property expressed by the following theorem and corollary.

Theorem 6. For all $0 < \alpha \leqslant 1$, the volume of E_α is less than the volume of E'.

Corollary 4. The volume of E_α is strickly decreasing on [0,1].

This corollary has provided an important result concerning the volume of the convex combination of linear form ellipsoids. It provides a method for strickly decreasing the volume of an ellipsoid while fixing the ellipsoid at a set of points. The next section uses this method in analyzing minimum spanning ellipsoids.

Minimum Spanning Ellipsoids - Brute Force

Given a set P, $|P|=n$, the MSE of P is the smallest (volume) spanning ellipsoid of P. It is denoted by E_P^*. In order to describe a technique for computing such an ellipsoid, we must use the structure of ellipsoids in E^d to analyze the number of conditions necessary for defining an ellipsoid.

Consider the centered form equation for an ellipsoid in E^d. How many degrees of freedom does the centered form equation have? Clearly in E^d, $\vec{x}_0$ consists of d independent variables. Moreover, A is positive-definite (symmetric), thus, A has $\frac{d(d+1)}{2}$ independent variables. Hence, $\frac{d^2+3d}{2}$ conditions uniquely determine an ellipsoid. There are sets of $\frac{d^2+3d}{2}$ points which cannot have any ellipsoid go through them. They determine other analytic objects (hyberboloids, paraboloids, etc.).

Given p points, $p<\frac{d^2+3d}{2}$, there can be many ellipsoids which go through the points. Specifically, we now define the *smallest* (in volume) such ellipsoid.

Definition 3. Given p points, $d+1 \leqslant p \leqslant \frac{d^2+3d}{2}$, in E^d, a *p-point ellipsoid* is defined to be the smallest ellipsoid which goes through the points.

The general method for computing a p-point ellipsoid will be to substitute the points into the centered or linear form equation and solve for as many independent variables as possible. Then, the volume will be minimized with respect to the remaining independent variables. In lower dimensions, a closed form equation can be developed for p-point ellipsoids. In higher dimensions, some sort of numerical method should be used to compute them.

Definition 3 gives a lower bound for p of $d+1$. The reason for this is apparent. In E^d, if $p \leqslant d$, then the set of p points are co-hyperplanar. Thus, the smallest ellipsoid through the points will have volume equal to zero. Similarily, we are assuming that our sets of points are not co-hyperplanar. This restriction merely keeps the problem well-formed.

The following theorem is the most important one of this section. It will provide the basic result necessary for developing a brute force algorithm for computing the MSE of a set of points. Moreover, it is used as the basis for the faster algorithm for computing the MSE.

Theorem 7 Given a set of points, P, in E^d, the minimum spanning ellipsoid of P, E_P^*, is a p-point ellipsoid.

The proof of this theorem uses a technique of shrinking a spanning ellipsoid, while fixing it at the points already on it. Eventually, either the ellipsoid becomes the smallest through those points (a p-point ellipsoid) or another point comes on it. Repeating this process proves the theorem. This technique is an important one, and will be used again in this paper.

Theorem 8. The minimum spanning ellipsoid of a set of points is unique.

This theorem now allows us to give the brute force algorithm for computing the mse of a set of points. It shows that the MSE can be computed by identifying some unique set of p points, for $(d+1) \leqslant p \leqslant \frac{d^2+3d}{2}$.

Algorithm B

[1] Compute every p-point ellipsoid for $d+1 \leqslant p \leqslant \frac{d^2+3d}{2}$.

[2] For each of the ellipsoids generated by [1], check if it spans the set of points.

[3] Choose the smallest which spans the points.

The asymptotic analysis shows that the runtime of this brute force algorithm is exponential in the dimension. It takes $O(n)$ time to perform step [2] for each of the $O(n^{\frac{d^2+3d}{2}})$ ellipsoids generated by [1]. Therefore, the runtime of Algorithm B is $O(n^{\frac{(d^2+3d+2)}{2}})$. This shows that the runtime of the brute force algorithm is exponential with respect to the dimension. This exponential runtime shows that in order to use MSE's in higher dimensions, a faster algorithm to compute them must first be developed.

A Fast Minimum Spanning Ellipsoid Algorithm

We now introduce an algorithm to compute the minimum spanning ellipsoid of a set of n points in $O(n^2)$ time. The algorithm is an iterative one. Given a p-point spanning ellipsoid, we will identify a smaller q-point spanning ellipsoid of the entire set, for some $d+1 \leqslant q \leqslant \frac{d^2+3d}{2}$. In the process, we will be able to eliminate a point from consideration. Thus, the algorithm is as follows:

Algorithm MSE

[1] Compute an initial p-point spanning ellipsoid of the entire set of points, for some $d+1 \leqslant p \leqslant \frac{d^2+3d}{2}$. This initial ellipsoid is called the *current ellipsoid.*

[2] Determine if there is a smaller spanning ellipsoid. If there is'nt one, then we are done. The current ellipsoid is the MSE.

[3] Eliminate a single point on the current ellipsoid which cannot be on the MSE.

[4] Identify a smaller q-point spanning ellipsoid, $d+1 \leqslant q \leqslant \frac{d^2+3d}{2}$, and go back to [2].

The iterative nature of this algorithm is similar to that in [Po82]. Step one, which initializes the algorithm, can be done in $O(n)$ time. Determining whether or not the current ellipsoid is minimum is shown to be decidable in constant time which depends on the dimension, independent of n. We present an algorithm to identify a smaller spanning ellipsoid in $O(n)$ time. Finally, a point can be eliminated by a constant number of applications of this algorithm. This analysis shows that the algorithm makes at most $O(n)$ iterations, yielding a worst-case runtime of $O(n^2)$.

In proving this algorithm correct, we will consider steps two, three, and four first, and then show how to compute an initial p-point spanning ellipsoid. For the following discussion, let E' be the current p-point spanning ellipsoid of the entire set of points P, and let Q be the set of p points on E'.

Theorem 9. There is a smaller spanning ellipsoid of the entire set P if and only if there is a smaller $(p-1)$-point spanning ellipsoid of Q.

This theorem provides the technique for determining if the current spanning ellipsoid, E', is minimum. We merely have to check to see if there is a $(p-1)$-point spanning ellipsoid of only the points on E'. There are exactly p such ellipsoids to check, thus, to do this merely requires $O(p)$ runtime. $p \leqslant \frac{d^2+3d}{2}$, therefore, we see that the runtime of this is only depends on the dimension, and is independent of the number of points in the set.

In order to show how to compute a smaller spanning ellipsoid of P in $O(n)$ time, we will use the technique of shrinking an ellipsoid while fixing it at a set of points introduced by Theorem 7. From Theorem 9, if the current ellipsoid is *not* minimum, then there exists a smaller q-point ellipsoid, $q=(p-1)$, of just the points on the current ellipsoid. Let this ellipsoid be called E, and let our ellipsoids be represented in their linear forms.

Algorithm ES (Ellipsoid Shrink)

[1] Does E span the entire set? If yes then we are done (E is smaller than E', i.e., smaller than the original p-point ellipsoid)

[2] Consider the convex combination of E and E', E_α. For each point *not* spanned by E, compute α such that E_α goes through the point. Let β be the minimum such α. Spanning is a convex property, thus E_β spans the entire set. From Theorem 6, this ellipsoid must be smaller than E'.

[3] This minimum value, identifies a point $\vec{x}$ such that E_β goes through $\vec{x}$ and the q points on E.

[4] Let E' be E_β, and let E now be the smallest ellipsoid through the $(q+1)$ points on E_β ($\vec{x}$ and the q points from the previous E). Iterate (loop to step [1]).

Each iteration of this algorithm shrinks a spanning ellipsoid through q points until it has another point from the set on it, while maintaining the property that it spans the entire set. At most $\frac{d^2+3d}{2}$ points determine an ellipsoid, thus, the algorithm iterates at most

$\frac{d^2+3d}{2} - p$ times (independent of n). Steps one, two, and three take $O(n)$ time. If we let $ME_d(p)$ represent the cost of computing a p-point ellipsoid in dimension d, then step four takes $O(ME_d(q+1))$ time, i.e., independent of n. Thus, this algorithm takes $O(n)$ time.

This algorithm has an important property expressed by the following theorem. Consider the initial value of E in Algorithm ES. It is a q-point spanning ellipsoid of the just the points on the current ellipsoid.

Theorem 9. The output of Algorithm ES, a spanning ellipsoid of P smaller than E', is the smallest spanning ellipsoid of P which goes through the q points on the initial ellipsoid E.

This theorem is important when considering step three of Algorithm MSE, eliminating a point from consideration. Let E' be the current spanning ellipsoid of the set, and Q be the set of points on E'.

Theorem 10. If E' is *not* minimum, then one of the points in Q cannot be on the mse.

Theorem 11. If a point in Q is on the minimum spanning ellipsoid of P, then it is on the smallest spanning ellipsoid of P which goes through at least $d+1$ of the points in Q.

This theorem provides the method by which we can eliminate one point from consideration. Given the current ellispoid, we compute, by brute force, all possible q-point spanning ellipsoids of just the points in Q, $q \geqslant d+1$. There are at most $O\left(\left(\frac{d^2+3d}{2}\right)^{\frac{d^2+3d}{2}}\right)$ of them. For each, use Algorithm ES to find in $O(n)$ time the smallest spanning ellipsoid of P which goes through the q points. Choose the smallest such one and eliminate a point from Q which is not on this ellipsoid. Thus, step three takes $O(n)$ time, however, the constant factor is exponential in the dimension.

We now introduce a technique for computing an initial p-point spanning ellipsoid of the entire set of points. This method uses the same iterative technique of the previous section. Given a spanning ellipsoid of P, E', through p points ($p > d$), let E be the smallest ellipsoid through those p points. We will now shrink E' as in Algorithm ES until either it is the smallest ellipsoid through those points or until another point is on it. This continues until there are at most $\frac{d^2+3d}{2}$ points on the ellipsoid. The only difference between this algorithm and Algorithm ES is that we have to identify this initial spanning ellipsoid E'. Once this ellipsoid is computed, we can then use Algorithm ES on it.

The rest of this section concerns itself with finding a spanning ellipsoid through $(d+1)$ points. As mentioned before, this spanning ellipsoid is not necessarily the smallest ellipsoid through the points. The initial spanning ellipsoid will in fact be a hypersphere, and we shall find such a spanning hypersphere in $O(n)$ time.

Consider the linear form for a hypersphere in E^d. That is, a hypersphere S, is an ellipsoid E: $(I, \vec{b}, c)$, where I is the identity matrix. This shows that a hypersphere is uniquely determined by $(d+1)$ non-hyperplanar points. Moreover, the unique hypersphere through some such set of $(d+1)$ points can be computed by solving the $(d+1)\times(d+1)$ set of linear equations generated by the linear form for S.

We now describe the algorithm to compute a spanning hypersphere of P. This hypersphere, denoted by $S_P(d)$, will go through $(d+1)$ non-hyperplanar points. As mentioned above, this hypersphere can be used as input to Algorithm ES to find an initial p-point spanning ellipsoid of the entire set.

Algorithm SH (Spanning Hypersphere)

[1] Identify a face of the convex hull of P. This face has at least d points, and at most $n-1$ points. Call the set of points on the face P'.

[2] Compute $S_{P'}(d-1)$, that is, a spanning hyperpshere in dimension $(d-1)$ of the face, P', of the convex hull found in step 1.

[3] $S_{P'}(d-1)$ goes through d points on the face found from step 1. Find the point not on the face which when added to the d points found by step 2, forms the largest hypersphere. This hypersphere goes through $(d+1)$ non-hyperplanar points, and spans the entire set.

Steps one and three can be done in $O(d \cdot n)$ time. Using the notation $T_d(n)$ to represent the asymptotic runtime of computing the spanning hypersphere of n points in dimension d, step two shows that

$$T_d(n) = T_{d-1}(n-1) + d \cdot n = O(d^2 \cdot n).$$

Note that this analysis ignores the cost of computing a hypersphere through a set of points, i.e., solving a set of linear equations. Thus, the actual asymptotic runtime is $O(d^2 \cdot M(d) \cdot n)$, where $M(d)$ is the asymptotic cost of multiplying $d \times d$ matrices.

Conclusions and Further Research

This concludes the description of the faster minimum spanning ellipsoid algorithm. As noted above, the runtime of the algorithm is $O(n^2)$. The dimensional costs are bounded above by the cost of eliminating a point as in step three. This constant is independent of n, yet is exponential in the dimension. One possible area of for further research would be to reduce this dimensional cost to a constant which is polynomial in the dimension. This could make minimum spanning ellipsoids useful for multi-dimensional optimization problems, such as linear programming.

Another area towards which further research should be directed is the problem of actually computing the smallest ellipsoid through p points. In [Po82], it is shown that closed form formulae for three-, four-, and five-point ellipsoids in two dimensions can be determined using differtial calculus. In higher dimensions, the algebraic manipulations necessary to do this become overwhelming. Use of symbolic algebra manipulators, such as MACSYMA, can accomplish this. However, in very high dimensions, some sort of numerical method, such as Newton's method, should be used to iteratively compute the desired ellipsoid.

References

[CK] Chand, D.R. and Kapur S.S., An Algorithm for Convex Polytopes, *J. ACM 17*, 1 (Jan. 1970).

[DR] D.P. Dobkin and S.P. Reiss, *The Complexity of Linear Programming*, Theoretical Computer Science, 11, (1980), pp. 1-18.

[FvD] J.D. Foley and A. van Dam, *Fundamentals of Interactive Computer Graphics*, Addison-Wesley Publishing Company Inc., Reading, 1982.

[GL] P. Gacs and L. Lovasz, *Khachian's Algorithm for Linear Programming*, Computer Science Department, Stanford University, 1979.

[G] B. Grünbaum, *Convex Polytopes*, Interscience Publishers, London, 1967.

[H] P. Halmos, *Finite Dimensional Vector Spaces*, Springer-Verlag, New York, 1974.

[MaO] Marhall, A.W., and Olkin, I., *Inequalities: Theory of Majorization and Its Applications*, Academic Press, Inc., New York, 1979.

[Po81] M. Post, *A Minimum Spanning Ellipse Algorithm*, Proc. 22nd IEEE Symposium on Foundations of Computer Science, October 1981.

[Po82] M. Post, *Computing Minimum Spanning Ellipses*, Brown University Technical Report No. CS-82-16, Providence, May 1982.

[Po83] M. Post, *Ellipsoids and Computational Geometry*, PhD thesis, Brown University, in submissison, 1983.

[S] M. Shamos, *Computational Geometry*, PhD thesis, Yale University, 1978.