

Reliable computation of the points of intersection of n spheres in \mathbb{R}^n

I.D. Coope *

(Received 7 August 2000)

Abstract

The problem of determining the points of intersection of n spheres in \mathbb{R}^n has many applications. Examples in 3-D include problems in navigation, in positioning of specific atoms in crystal structures, in reconstructing torso geometries in experimental cardiology, in the ‘Pentacle Problem,’ and in many other problems of distance geometry. The problem is easily formulated as a system of n nonlinear equations

*Department of Mathematics & Statistics, University of Canterbury, Private Bag 4800, Christchurch, NEW ZEALAND. <mailto:I.Coope@math.canterbury.ac.nz>

⁰See <http://anziamj.austms.org.au/V42/CTAC99/Coo1> for this article and ancillary services, © Austral. Mathematical Soc. 2000. Published 27 Nov 2000.

in the coordinates of the unknown point(s) of intersection and it is of interest to determine an efficient and reliable method of solution. It is shown that apart from a few square roots the problem is usually easily and robustly solved without iteration by employing standard techniques from linear algebra. In some applications, however, the radii of the spheres may not be known accurately and this can lead to difficulties, particularly when the required point is close to lying in the affine subspace defined by the n centres of the spheres. In such cases it is more appropriate to formulate a nonlinear least squares problem in order to identify a ‘best approximate solution.’ The special structure of this nonlinear least squares problem allows a solution to be calculated through an efficient safeguarded Newton iteration.

Contents

1	Introduction	C463
2	Solution by Gaussian Elimination	C464
3	Solution by Orthogonal Decomposition	C466
4	The Effects of Errors	C469
5	Discussion	C474
	References	C476

1 Introduction

The problem of determining the points of intersection of n spheres in \mathbb{R}^n has many applications. Examples in 3-D include problems in navigation, in positioning of specific atoms in crystal structures, in reconstructing torso geometries in experimental cardiology, in the ‘Pentacle Problem,’ and in many other problems of distance geometry (see, for example, [4]). For example, if the coordinates of n points in \mathbb{R}^n are known and it is required to calculate the coordinates of an unknown point when the distances of the unknown point from the given points are known then this problem is clearly equivalent to finding the intersection point(s) of n spheres in \mathbb{R}^n . This interpretation allows an easy geometric proof that usually there will be two points of intersection, because, if $\mathbf{x} \in \mathbb{R}^n$ is a solution to the problem, then clearly the reflection of \mathbf{x} in the affine subspace defined by the n given points will also be a solution. The Pentacle Problem (see, for example, Mackay [6]) is a slight variation of this latter problem. In this case there are two unknown points and it is required to find the distance between them when the distances from each unknown point to the n known points are given. Of course, there are two solutions to this problem too, depending on whether or not the two unknown points lie on the same side or opposite sides of the affine subspace through the n given points.

Let $\mathbf{a}_j \in \mathbb{R}^n$, $j = 1, 2, \dots, n$, denote the centres of the n ‘spheres’ and d_j , $j = 1, 2, \dots, n$, the corresponding radii. Then the intersecting n -spheres

problem is to find $\mathbf{x} \in \mathbb{R}^n$ satisfying the n nonlinear equations:

$$\|\mathbf{x} - \mathbf{a}_j\|_2^2 = d_j^2, \quad j = 1, 2, \dots, n, \quad (1)$$

or equivalently,

$$\mathbf{x}^T \mathbf{x} - 2\mathbf{x}^T \mathbf{a}_j + \mathbf{a}_j^T \mathbf{a}_j = d_j^2, \quad j = 1, 2, \dots, n. \quad (2)$$

If the spheres do not intersect then the Equations (1) will have no solution and it may be important to be able to detect this efficiently. Sometimes the radii of the spheres will correspond to distances, as in some of the earlier examples. Then, even if there is no point of intersection, it may be required to determine a ‘best approximation’ in the sense of making the residuals of the Equations (1) small in some appropriate norm. This aspect is considered again in Section 4. In Sections 2 and 3 solutions via Gaussian elimination and via orthogonal decomposition are considered when it is known from other considerations that a solution is guaranteed to exist.

2 Solution by Gaussian Elimination

Let $\mathbf{A} \in \mathbb{R}^{n \times n}$ be the matrix whose columns are the vectors \mathbf{a}_j , $j = 1, 2, \dots, n$. If the vectors $\{\mathbf{a}_j\}_1^n$ are linearly independent then this matrix is non-singular and the following approach provides a simple solution technique. First rewrite the nonlinear equations (2) as

$$\mathbf{a}_j^T \mathbf{x} = (r + b_j)/2, \quad j = 1, 2, \dots, n, \quad (3)$$

where $r = \mathbf{x}^T \mathbf{x}$ and $b_j = \mathbf{a}_j^T \mathbf{a}_j - d_j^2$, $j = 1, 2, \dots, n$. In matrix form, Equations (3) become $\mathbf{A}^T \mathbf{x} = (r\mathbf{e} + \mathbf{b})/2$, or

$$\mathbf{x} = (r\mathbf{u} + \mathbf{v})/2, \quad (4)$$

where $\mathbf{e} \in \mathbb{R}^n$ denotes the vector $\mathbf{e} = [1, 1, \dots, 1]^T$ and

$$\mathbf{u} = \mathbf{A}^{-T} \mathbf{e}, \quad \mathbf{v} = \mathbf{A}^{-T} \mathbf{b}. \quad (5)$$

Hence, $r = \mathbf{x}^T \mathbf{x} = \frac{1}{4}(r\mathbf{u} + \mathbf{v})^T (r\mathbf{u} + \mathbf{v})$ or

$$(\mathbf{u}^T \mathbf{u})r^2 + (2\mathbf{u}^T \mathbf{v} - 4)r + \mathbf{v}^T \mathbf{v} = 0, \quad (6)$$

which is a quadratic equation in the scalar, r . Solving for r gives,

$$r = \frac{2 - \mathbf{u}^T \mathbf{v} \pm \sqrt{(2 - \mathbf{u}^T \mathbf{v})^2 - (\mathbf{u}^T \mathbf{u})(\mathbf{v}^T \mathbf{v})}}{\mathbf{u}^T \mathbf{u}}, \quad (7)$$

and the two solutions for \mathbf{x} can then be recovered using Equation (4).

The above approach is efficient, requiring the solution of two linear systems of Equations (5) of order n . Because each system involves the coefficient matrix \mathbf{A}^T only one matrix factorization (LU decomposition) is required and the overall cost (ignoring $O(n)$ computations) is about

$$\frac{1}{3}n^3 + n^2 \quad (8)$$

multiplications (with a similar number of additions) and only one square root. Unfortunately, it is not sufficiently robust as a little thought reveals.

First, it is entirely plausible that the origin will be one of the n given points, resulting in a zero column in the matrix \mathbf{A} . Second, in many applications the n given points will be at the same ‘height,’ which could result in a zero row in \mathbf{A} . Of course, both situations may exist with \mathbf{A} having both a zero row and column (and there are other ways that \mathbf{A} may be singular too). In most applications, though, \mathbf{A} will have $\text{rank}(\mathbf{A}) \geq n - 1$ because the given points \mathbf{a}_j , $j = 1, \dots, n$ will be affinely independent, so a possible quick remedy in this case, (when $\text{rank}(\mathbf{A}) = n - 1$), is to find a shift of origin which makes \mathbf{A} non-singular. However, a different approach is preferred which always reduces the original problem to an equivalent one in which the matrix \mathbf{A} has a zero row and column.

3 Solution by Orthogonal Decomposition

A more robust method of solution is to apply a translation and rotation/reflection of axes and then work in the transformed space. First, shift the origin to one of the given centres. The point corresponding to the smallest d_j is recommended since the required solution is closest to this point. For convenience it is assumed that this is the last point \mathbf{a}_n . Let $\hat{\mathbf{A}}$ denote the $n \times (n - 1)$ matrix of shifted points:

$$\hat{\mathbf{A}} = [\mathbf{a}_1 - \mathbf{a}_n, \mathbf{a}_2 - \mathbf{a}_n, \dots, \mathbf{a}_{n-1} - \mathbf{a}_n]. \quad (9)$$

Now apply an orthogonal transformation that annihilates all entries in the last row of $\hat{\mathbf{A}}$. This is most conveniently done by calculating an orthogonal

decomposition (QR factorization) of $\hat{\mathbf{A}}$. Thus

$$\hat{\mathbf{A}} = \mathbf{Q} \begin{bmatrix} \mathbf{R} \\ \mathbf{0}^T \end{bmatrix}, \quad (10)$$

where \mathbf{Q} is a $n \times n$ orthogonal matrix and \mathbf{R} is a $(n-1) \times (n-1)$ upper triangular matrix and now \mathbf{x} can be written as

$$\mathbf{x} = \mathbf{Q} \begin{bmatrix} \mathbf{y} \\ z \end{bmatrix} + \mathbf{a}_n \quad (11)$$

where $\mathbf{y} \in \mathbb{R}^{n-1}$ and $z \in \mathbb{R}$. The advantage of this transformation is that the operations of translation and rotation/reflection preserve Euclidean length so that Equations (1) can then be written

$$\|\mathbf{y} - \mathbf{r}_j\|_2^2 + z^2 = d_j^2, \quad j = 1, 2, \dots, n-1, \quad (12)$$

and

$$\|\mathbf{y}\|_2^2 + z^2 = d_n^2. \quad (13)$$

where \mathbf{r}_j denotes the j th column of \mathbf{R} . Using Equation (13) to replace the nonlinear terms in Equations (12) by d_n^2 then gives the system of linear equations,

$$\mathbf{R}^T \mathbf{y} = \mathbf{c}, \quad (14)$$

where $\mathbf{c} \in \mathbb{R}^{n-1}$ has components $c_j = \frac{1}{2}(d_n^2 - d_j^2 + \|\mathbf{r}_j\|_2^2)$. The linear system (14), is easily solved by forward substitution for the vector \mathbf{y} and then (13) gives

$$z = \pm \sqrt{d_n^2 - \|\mathbf{y}\|_2^2} \quad (15)$$

and the required points of intersection are then determined by applying the transformation (11).

The computational cost of this approach is a little higher than the method of the previous section. Ignoring computations of $O(n)$ the cost is about n^2 subtractions for $\hat{\mathbf{A}}$, followed by $\frac{2}{3}n^3$ multiplications and additions for \mathbf{Q} and \mathbf{R} . Then $\frac{1}{2}n^2$ for \mathbf{y} followed by a further n^2 to recover \mathbf{x} . In all a total of about

$$\frac{2}{3}n^3 + \frac{5}{2}n^2 \quad (16)$$

multiplications and additions and n square roots if, for example, the \mathbf{QR} factors are calculated by Householder transformations. For large n this is about twice as much work as the method of Section 2. In practice, many applications occur with $n = 3$ and then the extra scalar products and vector additions required by the method of Section 2 actually make the method of this section preferable from both the viewpoints of efficiency, and numerical accuracy and stability. The amount of computation for the method of this section can also be reduced further by choosing \mathbf{Q} to annihilate only the elements of the last row of $\hat{\mathbf{A}}$ (using Givens rotations, for example). Then \mathbf{R} would no longer be upper triangular and the main cost of the computation would be $\frac{1}{3}(n-1)^3$ multiplications/additions in solving Equation (14) by Gaussian elimination. Thus the asymptotic cost for the method of this section could also be reduced to $\frac{1}{3}n^3 + O(n^2)$ as for the method of Section 2.

The method of this section, however, has the added advantage of revealing, precisely, the conditions under which solution(s) to the problem exist.

4 The Effects of Errors

In this section we assume that the points \mathbf{a}_j , $j = 1, 2, \dots, n$ are affinely independent (this should be the case in all practical applications) so that the matrix \mathbf{R} in Equation (14) will have full rank and \mathbf{y} is uniquely defined. Thus, provided that the calculated solution satisfies

$$\|\mathbf{y}\|_2 \leq d_n, \quad (17)$$

there will always be a solution to the problem. Moreover, it is easily seen that the spheres will have one point of intersection if and only if $\|\mathbf{y}\|_2 = d_n$, in which case $z = 0$ and the required point lies in the affine subspace defined by the centres of the n n -spheres. However, in practical applications the distances d_j , $j = 1, 2, \dots, n$ may correspond to measurements and there will be errors.

The effect of errors can be particularly severe when the required point is close to lying in the affine subspace defined by the given centres of the spheres as the following example shows. Consider the case where $n = 3$ and the centres of the spheres are the columns of the matrix

$$\mathbf{A} = \begin{bmatrix} 9 & 9 & 1 \\ 1 & 2 & 3 \\ 8 & 6 & 3 \end{bmatrix}$$

and the true distances (radii of the spheres) are the elements of the vector

$$\mathbf{d} = [5.8518, 7.0837, 8.2641]^T,$$

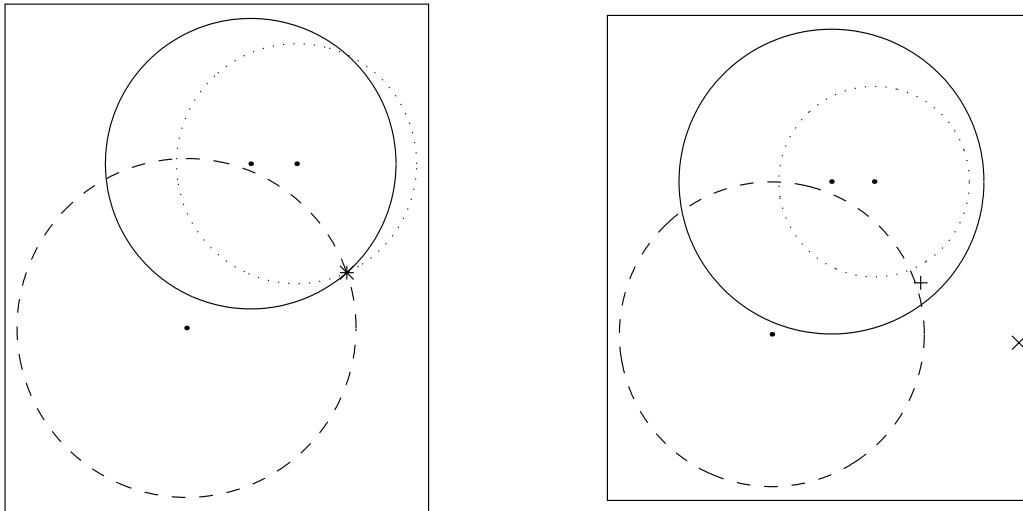


FIGURE 1: Cross-section of 3 spheres intersecting at a point (left) and not intersecting (right).

but the measured distances are given by the elements of the vector $\hat{\mathbf{d}}$ where

$$\hat{\mathbf{d}} = [5, 8, 8]^T.$$

The elements of \mathbf{d} were chosen so that the spheres intersect at a single point (to 4 d.p. accuracy) and a cross-section of the spheres taken through the plane defined by their centres is displayed in Figure 1(left). (Note that this plane corresponds to setting $z = 0$ in (11).) In Figure 1(right), the centres remain the same as in Figure 1(left), but now the radii are given by the elements of the vector $\hat{\mathbf{d}}$ (the errors have been exaggerated for extra visual effect). It can be seen that the effect of decreasing the radius of the first sphere and increasing the radius of the second sphere is to cause the first sphere to lie completely inside the second so that there can be no points of intersection for the three spheres.

Intuitively, a good approximation to the required point (accepting that errors have been made in the measurements) would lie in the affine subspace defined by the centres of the spheres somewhere near the point marked ‘+’ in Figure 1(right). Of course, an easy way to generate an ‘approximate solution’ in such a case is simply to argue that because the negative value of z^2 is due to errors in the measurements of d_j , $j = 1, \dots, n$, it is reasonable to set $z = 0$ in the method of calculation outlined in Section 3. Unfortunately, this simple device does not always provide an acceptable approximation. For the example given above, setting $z = 0$ results in the computed approximation to the ‘point of intersection’ as the point indicated by ‘×’ in Figure 1(right). Clearly, this point is unacceptable.

The point ‘+’ is, in fact, the solution to the nonlinear least squares problem

$$\min S(\mathbf{x}), \quad \mathbf{x} \in \mathbb{R}^n, \quad \text{where} \quad S(\mathbf{x}) = \sum_{j=1}^n \{\|\mathbf{x} - \mathbf{a}_j\|_2 - d_j\}^2, \quad (18)$$

which explains why this gives a visually acceptable approximation. It is easy to see that the solution to problem (18) lies in the appropriate affine subspace (corresponding to $z = 0$) by inspecting the gradient vector of $S(\mathbf{x})$.

$$\nabla S(\mathbf{x}) = 2 \sum_{j=1}^n (\mathbf{x} - \mathbf{a}_j) \alpha_j, \quad (19)$$

where

$$\alpha_j \equiv \alpha_j(\mathbf{x}) = \left(1 - \frac{d_j}{\|\mathbf{x} - \mathbf{a}_j\|_2}\right).$$

Therefore

$$\nabla S(\mathbf{x}) = \mathbf{0} \iff \mathbf{x} \sum \alpha_j = \sum \alpha_j \mathbf{a}_j, \quad (20)$$

and hence, if $\alpha_j \neq 0$, $j = 1, 2, \dots, n$, then \mathbf{x} must lie in the affine subspace defined by the points \mathbf{a}_j , $j = 1, \dots, n$. Although problem (18) is very similar in structure to the ‘circle fitting problem’ discussed in [3], the transformations that were successful in reducing this latter problem to a linear least squares problem do not work in the current context and problem (18) is essentially nonlinear and must be solved by iterative techniques. However, the results of Section 3 are still useful in reducing the amount of computation of this nonlinear least squares problem because it is known that the solution has

$z = 0$. Therefore, it is only required to find the nonlinear least squares solution for $\mathbf{y} \in \mathbb{R}^{n-1}$. Thus the iterative method can be applied to the following problem in $n - 1$ variables.

$$\min S(\mathbf{y}), \quad \mathbf{y} \in \mathbb{R}^{n-1}, \quad \text{where} \quad S(\mathbf{y}) = \sum_{j=1}^n \{\|\mathbf{y} - \mathbf{r}_j\|_2 - d_j\}^2, \quad (21)$$

where \mathbf{r}_j , $j = 1, 2, \dots, n - 1$, denotes the j th column of \mathbf{R} , as before, and $\mathbf{r}_n \equiv \mathbf{0}$. Although this is only a small saving, it is still worthwhile and the solution to Equation (14) gives a suitable initial approximation. Of course, this problem is only solved when $\|\mathbf{y}\|_2 > d_n$ because the original problem has a well-defined solution otherwise. First and second derivatives of $S(\mathbf{y})$ defined by (21) are easily computed,

$$\nabla S(\mathbf{y}) = 2 \sum_{j=1}^n (1 - \gamma_j)(\mathbf{y} - \mathbf{r}_j), \quad (22)$$

where γ_j denotes the scalar

$$\gamma_j \equiv \gamma_j(\mathbf{y}) = d_j / \|\mathbf{y} - \mathbf{r}_j\|_2,$$

and

$$\nabla^2 S(\mathbf{y}) = 2 \left[\mathbf{I} \sum_{j=1}^n (1 - \gamma_j) + \sum_{j=1}^n \gamma_j \mathbf{w}_j \mathbf{w}_j^T \right]. \quad (23)$$

where \mathbf{w}_j is the unit vector,

$$\mathbf{w}_j = (\mathbf{y} - \mathbf{r}_j) / \|\mathbf{y} - \mathbf{r}_j\|_2, \quad j = 1, 2, \dots, n. \quad (24)$$

Inspection of the form for $\nabla^2 S(\mathbf{y})$ also reveals that this matrix is certainly positive definite for all values of \mathbf{y} satisfying $\|\mathbf{y} - \mathbf{r}_j\|_2 > d_j$, $j = 1, 2, \dots, n$, and this region always includes the recommended starting point. Thus a safeguarded Newton method (see, for example, Fletcher [5]) is particularly well-suited. This was the approach taken for solving the problem illustrated in Figure 1(right). Usually, good accuracy is obtained in 2 or 3 iterations.

5 Discussion

Techniques for solving the nonlinear equations arising from the intersecting n -spheres problem have been described and compared and apart from a few square roots the problem has been shown to be most effectively solved by employing standard techniques from linear algebra. Unlike many realistic applications it is a problem that requires very little by way of introduction and is easily described mathematically. In particular the problem provides an excellent example for instructors of undergraduate linear algebra (& applications) courses since it allows the concepts of linear independence, affine independence, affine sub-spaces, and orthogonal transformations to be employed in a context which allows for easy geometric interpretation. All the calculations and diagrams of this paper were produced very easily using MATLAB which is available in most university computing environments.

Although the problem arises most naturally in three dimensions, the solution techniques described apply to any finite dimensional case. In some

situations it may be preferable to ‘over-sample’ in order to assess more reliably (and quantitatively) the effects of errors. Then the problem becomes a non-linear least squares problem with $m > n$ measured distances from m given points in \mathbb{R}^n . The technique of Section 3 may still be used here to provide a ‘reasonable’ initial estimate, simply by choosing only n of the points in the first instance.

In navigational problems and problems relating to crystal structures it is usual to have measurements corresponding to bearings or angles as well as distances. In this situation the use of the orthogonal decomposition would no longer be successful in reducing the problem to a simple system of linear equations.

Finally, it should be noted that in the problem considered here there is only one point whose coordinates are unknown. Such problems frequently arise when measuring devices are fixed at known positions and observations are then made to determine the appropriate distances from the required unknown point. In many other problems (e.g. molecular conformation) the coordinates of all points are unknown but the distances between all possible points are known although usually with errors present. In this case the problem is sometimes referred to as the “fundamental problem of distance geometry” and this has received much recent interest. A good overview of this latter problem is given in [4] and more recent developments can be found in [1, 2, 7], and the references therein.

Acknowledgements: The author is grateful to Dr C.J. Price for comments and suggestions leading to improvements on an earlier version of this paper. The application to reconstruction of torso geometries in experimental cardiology was unknown to the author in the early stages of the preparation of this paper and was pointed out by Rob MacLeod of the Nora Eccles Harrison Cardiovascular Research and Training Institute, University of Utah.

References

- [1] S. Al-Homidan. *Hybrid Methods for Optimization Problems with Positive Semi-Definite Matrix Constraints*. PhD Thesis, Department of Mathematics and Computer Science, University of Dundee, Scotland, 1993. C475
- [2] S. Al-Homidan and R. Fletcher. Hybrid methods for finding the nearest Euclidean distance matrix. In *Recent advances in nonsmooth optimization*, pages 1–17, River Edge, NJ, 1995. World Publishing. C475
- [3] I.D. Coope. Circle fitting by linear and nonlinear least squares. *Journal of Optimization Theory & Applications*, 76.2:381–388, 1993. C472
- [4] G.M. Crippen and T.F. Havel. *Distance Geometry and Molecular Conformation*. Research Studies Press Ltd, John Wiley & Sons Inc., 1988. C463, C475

- [5] R. Fletcher. *Practical Methods of Optimization*. John Wiley & Sons, 2nd Edition, New York, 1987. C474
- [6] A.L. Mackay. Generalized structural geometry. *Acta Cryst.*, A30:440–447, 1974. C463
- [7] A.Y. Alfakih, A. Khandani, and H. Wolkowicz. Solving Euclidean distance matrix completion problems via semidefinite programming. *Computational Optimization and Applications*, 12:13–30, 1998. C475