The Protein Folding Problem and Distance Geometry

by Robert Reams, University of Kentucky.

The protein folding problem is an example of how matrix theory has come to have an important application in biology, in the form of distance matrices. Non-linear optimization is applied in combination with distance matrices to this problem, although the former will not be discussed here. In this exposition I hope the experts will forgive my stripping away almost all technical details.

A protein molecule is a connected sequence of amino acid molecules, and there are just twenty amino acids in nature (Figure 1 shows a fairly typical amino acid). By representing each amino acid by a letter from a twenty letter alphabet, we can say that a protein is a string of several hundreds or thousands of these letters. The problem of finding the ordered sequence of amino acids in a protein has been solved, in fact it has been automated with a sequenator; using enzymes, snip the full protein into strings of no longer than 100 amino acids, keeping track of the break-points, and the output from the sequenator gives you the sequence. Our problem goes farther than this: Determine the way in which a given protein has folded in on itself, knowing its amino acid sequence, to produce its three dimensional form. It is the protein's three dimensional shape which largely determines how a protein functions, how a protein acts as an enzyme in chemical reactions in the body, or how a protein behaves as an antibody in the immune system. Proteins lose their three dimensional structure on heating (cooking). Once cooked, enzymes in the stomach and small intestine further break the protein into its individual amino acids, and they are then incorporated into our bodies. There are many ways in which the biologist goes about attacking the incredibly important problem of determining a protein's three dimensional structure. Some examples are, x-ray crystallography, where it is possible to reconstruct from x-ray diffraction patterns an accurate picture of the molecule; molecular dynamics, which is a computer simulation of energy minimization, where each atom is made to obey Newton's equations of motion; or distance geometry algorithms. If the protein cannot be crystallized, which is a common occurrence, the molecular biologist would probably perform a combination of the last two. An intriguing aspect of the protein folding problem, which has yet to be understood, is that the amino acid sequence seems to completely specify how the protein folds.

In the 1930's, K. Menger, and I. V. Schoenberg (who later invented splines) initiated the area of distance geometry and the study of distance matrices. It has become an active area of research in the last fifteen years, partly because of the applications described here. It is still an area open to purely mathematical investigation, with

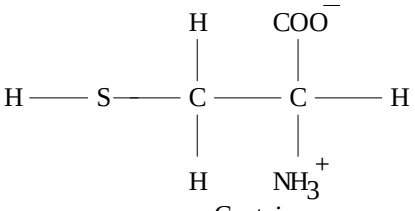
many problems where the solutions would be useful for the biologist, and for which there are large sums of money available in the form of grants.

A distance matrix is a matrix for which the (i, j)-entry is the distance, or more usually the square of the distance, between vertex i and vertex j, in a set of n vertices. Clearly, such a matrix has nonnegative entries, is symmetric, and has all zeroes down the diagonal. What is less obvious is deciding when such a matrix is a distance matrix, i.e. given a symmetric matrix with zeroes down the diagonal, and nonnegative entries, when does it correspond to a shape with n vertices in R^{n-1}. Clearly, again, for any three vertices, the square roots of the corresponding entries in the distance matrix must satisfy the triangle inequality. This is another necessary condition for a matrix to be a distance matrix, and to see that it is not a sufficient condition, try to draw a tetrahedron (not necessarily a regular tetrahedron) in three dimensions which corresponds to the matrix

$$\begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 4 \\ 1 & 1 & 0 & 1 \\ 1 & 4 & 1 & 0 \end{bmatrix}.$$

You will find yourself drawing two adjoining equilateral triangles, but you won't be able to form a three dimensional shape with the given lengths. Necessary and sufficient conditions for an $n \times n$ matrix to be a distance matrix, and the coordinates of the vertices of the shape in R^{n-1}, were given by Schoenberg.

If the biologist had all the interatomic distances in his or her protein, it would be a simple matter to fill in the entries of the distance matrix, and from Schoenberg's theorem calculate and portray on a computer screen the three dimensional structure. Unfortunately, if the protein cannot be crystallized, these interatomic distances can be difficult to determine. The x-ray crystal structure of each of the twenty amino acids in nature is known, however, and so the lengths of the bonds between their atoms are known. We also know the amino acid sequence of our protein, so we can fill in the entries of all the $k \times k$ blocks along the diagonal of the matrix which is to be our distance matrix, where k is the number of atoms in the amino acid corresponding to that block. Proteins also frequently form disulfide bonds with itself. That is to say, it often happens that two cysteine amino acids (see Figure 1), widely separated in the sequence of the protein, form a bond between their sulfur atoms. From the known bond length of these bonds, this would give us some distance matrix entries which are far from the diagonal. In Figure 1, the groups of atoms next to the + and − which denote ions, are common to all amino acids, and these are the points of contact with their adjacent amino acids. Most amino acids have similar side chains, although without sulfur atoms.



Cysteine

(C = Carbon, O = Oxygen, H = Hydrogen, N = Nitrogen, S = Sulfur)

Figure 1.

It is possible to find some interatomic distances between atoms which are not bonded, and which are in different amino acids, by performing a nuclear magnetic resonance (NMR) experiment. We place our protein in a strong (and I mean *really* strong) magnetic field, provided by a superconducting magnet. This has the effect of aligning a slight excess of the hydrogen nuclei in our protein in the direction of the magnetic field, the rest are aligned against the field. Radio waves (500 MHz), in pulse form, are then applied and detectors are placed around the protein to detect a resonance signal. According as different resonance signals are detected, molecules containing hydrogen atoms can be recognized from their distinctive spectrum (amino acids, surprise!). What is of interest for us, however, is that at a place where two hydrogen nuclei are situated as close together as 5 Ångstroms (about two or three bond lengths), and when the sample is irradiated in a certain way, a characteristic effect known as the nuclear Overhauser effect (NOE) is observed. It is in this way that we find more entries for our distance matrix, these entries connecting different diagonal blocks. This NOE information, as well as some other information from the NMR experiment, can enable us to identify some typical shapes within the protein. Commonly seen shapes within proteins are α-helices, which are right-handed helix shapes where the amino acids have wound around each other, or β-sheets, where groups of consecutive amino acids run parallel to other such groups. See Figure 2, which shows the main strand of the protein molecule BPTI. BPTI contains only 58 amino acids but an alpha helix is visible jutting out to the left of the figure, while a two strand β-sheet (β-sheets often contain five or more side by side strands) can be seen as a long loop in the bottom right of the figure. NOE information also enables us to detect a disulfide bond.

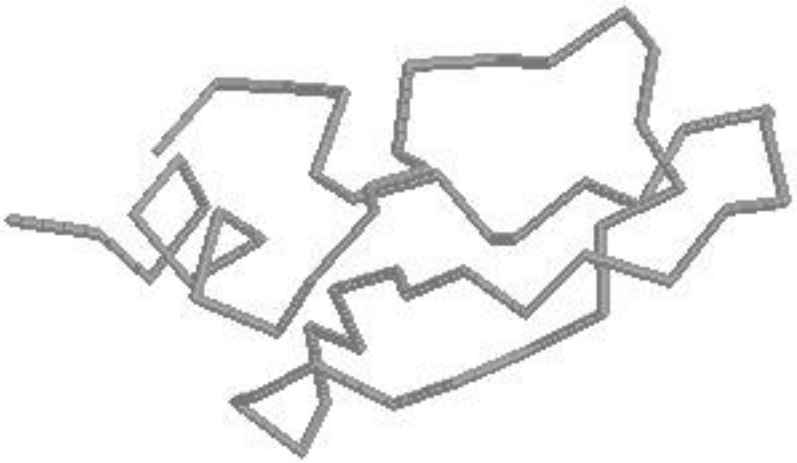


Figure 2.

At the end of all this we have a matrix with known entries (bond lengths and NOE distances), and unknown entries (for long distances). We need to now fill in the unknown entries so that the matrix is a distance matrix which corresponds to a shape in R^3. There might be more than one way to do this, since there is nothing to stop the protein flopping about in solution (the protein would be in some solution during the NMR experiment, usually water). We will content ourselves with finding a way to sample the set of all possible conformations.

With these unknown distances there are many ways to proceed, and only one

path will be described here, although it will have much in common with currently available software packages. The choice of how to proceed is mainly determined by how time-consuming the algorithm is to implement on a computer, particularly on a protein molecule with many amino acids. To repeat what has been said already, we have an $n \times n$ matrix, where n is the number of atoms in our protein, with many unknown entries, and we would like to fill in these entries in such a way that there is a shape in R^3 which corresponds to it.

We will first make sure that every 3×3 and 4×4 sub-matrix, across the diagonal of our matrix, which corresponds to a matrix of distances between every subset of 3 or 4 vertices, is a distance matrix. We do this since we know that it is a necessary condition for the full matrix to be a distance matrix. It is an easy exercise for the reader to check that three lengths can form the sides of a triangle if and only if all three triangle inequalities hold. For a tetrahedron, a quick sketch also shows that the triangle inequality must hold for the lengths between every triplet of vertices (although it does not necessarily hold for every three of the six lengths), which gives a necessary condition that the lengths form the sides of a tetrahedron. We saw from the 4×4 matrix given earlier in the text that this is not a sufficient condition to be able to construct a tetrahedron. The reader should also be easily persuaded that if two faces of a tetrahedron are made to stay joined by an edge, making this a hinge; then the edge joining the moving vertices of the two faces, can be no longer than a certain distance, and no shorter than a certain distance. These two extreme situations are achieved when the simplex lies flat in the plane, as in the two right-hand tetrahedra in Figure 3.

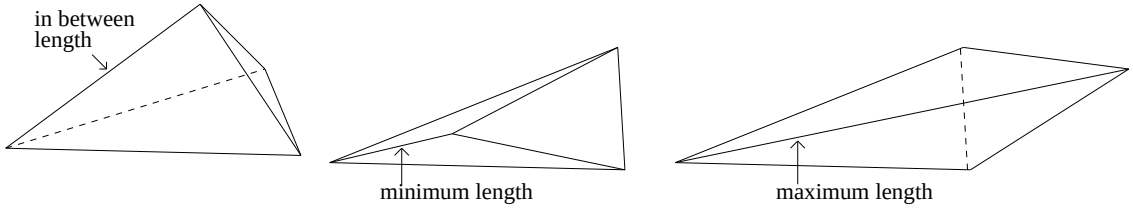


Figure 3.

If the length of the edge in question is between these two certain distances, it is said to satisfy the tetrangular inequality. Basic trigonometry, using the law of cosines, will give the (messy) form of this inequality. This inequality must hold for the length of any of the edges, keeping the other edges of fixed length. The triangle inequalities and tetrangular inequalities, together, give us necessary and sufficient conditions for the construction of a tetrahedron.

The triangle and tetrangular inequalities give us necessary upper and lower bounds, for the unknown distance entries of our full distance matrix. The lower bounds for the unknown distances are also partly determined by the fact that there are lower bounds for the distance between any two atoms, since two atoms can only come to within a certain minimum distance before their nuclei would repel each other. An upper bound for all interatomic distances can also be had, by just not letting the atoms be farther apart than the strung out length of the protein. By combining all of these upper and lower bounds for each (i, j)-entry, if we randomly generate a matrix which has entries between these bounds, the matrix is more likely to be a distance matrix than without any bounds (if we have an accurate bond-length

for a given entry, the upper bound equals the lower bound).

We next try to find the *closest* distance matrix to our randomly generated matrix. By "closest" we mean using the Frobenius norm, i.e. given two $n \times n$ matrices $A = (a_{ij})$ and $B = (b_{ij})$, the distance between A and B is given by $||A - B|| = \sqrt{\sum_{i,j=1}^{n}(a_{ij} - b_{ij})^2}$. There is a procedure, discovered by John von Neumann, to find the closest point in the intersection of two subspaces to a given point. This procedure has plausible picture reasoning, see Figure 4, where we project orthogonally onto one subspace, then the other, back and forth. If the two sets are convex there is an improvement of this algorithm due to Dykstra, to find the closest point in the intersection of two convex sets to a given point, using the same principle of alternately projecting on the two convex sets [3].

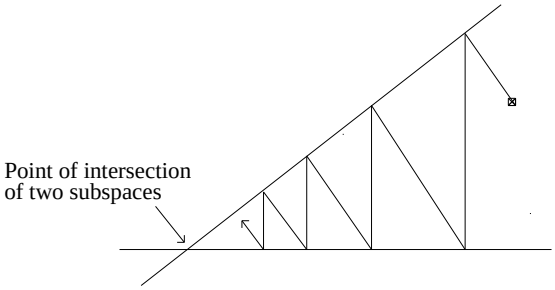


Figure 4.

It isn't an obvious geometric fact that the set of distance matrices is a convex set, although this is part of what Schoenberg proved: Let D be an $n \times n$ symmetric matrix with zeroes down the diagonal. D is a distance matrix if and only if $x^T D x \leq 0$, for all vectors x orthogonal to the vector $e = (1, 1, ..., 1)^T$. The value of this theorem for us is that the set of distance matrices is equal to the intersection of ... the convex set of symmetric matrices with zeroes down the diagonal, and the convex set of symmetric matrices which are negative semi-definite on the aforementioned $(n-1)$-dimensional subspace of R^n. We can then use Dykstra's algorithm to find the closest matrix in the intersection, by projecting alternately back and forth between these two sets. The trouble with this algorithm is that it is extremely time-consuming to implement. So after only several projections we stop, and convert to a shape in R^3 whose distance matrix, although not necessarily the closest to our original random matrix, we expect to be not far away.

In the interests of your time, my space, and an inclination to avoid technicalities in this account, I will skip the remainder of the argument showing how to convert to a shape with n vertices in R^3. Then the algorithm continues to look for the closest distance matrix in R^3, using some very efficient iterative methods from non-linear optimization. Further details on all of the above, can be best found in [2] and [4], for which I hope the reader might at least be curious.

No account of distance matrices would be worth even a grain of salt, without mentioning the next result. This is a result about distance matrices that the ancient Greeks seem to have missed, although it did not escape Schoenberg. Given a shape with n vertices in R^{n-1}, if you go around the shape calculating the square roots of the sides, it is a remarkable fact that these new lengths can form a shape with n vertices also! It is not a difficult exercise to show that if you are given a triangle, and

you take the square roots of the lengths of the sides, the new lengths can also form a triangle. Having done this it is just as easy to further show that if you take kth roots, for any positive integer k, you again can form a triangle. Schoenberg showed this result to be true for any real exponent α, where $0 < \alpha \leq 1$, i.e. let $D = (d_{ij})$ be an $n \times n$ distance matrix, then the matrix $\tilde{D} = (d_{ij}^{\alpha})$ is also a distance matrix [1] p.135.

Further Reading

[1] L. M. Blumenthal, *Theory and Applications of Distance Geometry*, Oxford University Press, Oxford, 1953. Reprinted by Chelsea Publishing Co., New York, 1970.
[2] G. M. Crippen and T. F. Havel, *Distance Geometry and Molecular Conformation*, Research Studies Press, Chemometrics Series, Vol. 15, 1988.
[3] W. Glunt, T. L. Hayden, S. Hong, and J. Wells, An Alternating Projection Algorithm for Computing the Nearest Euclidean Distance Matrix, *SIAM J. Matrix Anal. Appl.,* 11(4), 589–600, 1990.
[4] K. M. Merz, Jr. and S. M. Le Grand, Editors *The Protein Folding Problem and Tertiary Structure Prediction*, Birkhäuser, Massachusetts, 1994.