GEPOL: An Improved Description of Molecular Surfaces. III. A New Algorithm for the Computation of a Solvent-Excluding Surface

J. L. PASCUAL-AHUIR, E. SILLA,* and I. TUÑON

Departamento de Química Física, Universidad de Valencia, 46100 Burjassot (Valencia), Spain

Received 13 December 1993; accepted 21 March 1994

ABSTRACT

To understand and calculate the interactions of a solute with a solvent, a good method of computing the molecular surface is needed. Three kinds of surfaces may be used: the van der Waals Surface, the Accessible Surface, and the Molecular Surface. The latter is redefined in this article as the Solvent-Excluding Surface. The new algorithm for computing the Solvent-Excluding Surface included in the GEPOL93 program is described. GEPOL93 follows the same concept as former versions of GEPOL but with a full new algorithm. Thus, it computes the Solvent-Excluding Surface by filling the spaces not accessible to the solvent with a set of new spheres. The computation is controlled by three parameters: the number of triangles per sphere, controlled by NDIV; the maximum overlap among the new spheres (OFAC); and the size of the smallest sphere that can be created (RMIN). The changes introduced for the computation of the ESURF make GEPOL93 not just a new version but a new program. An estimation is made of the error in the area and volume obtained in the function of the parameters. © 1994 by John Wiley & Sons, Inc.

Introduction

With the development of computers in the last two decades, the concept of molecular surface has become more common. Thus, the surface has been used in microscopic models of solution.[1-4] Linear relations have also been found between molecular surface and solvation energy.[5-8] Moreover, application of the molecular surface concept has become popular in the world of structural biophysics. Thus, given that molecular surfaces can help us in calculating the interactions of a molecule with its surroundings, they are one of the main tools in understanding the folding and stability of proteins.[9] This concept has also been used in the study of docking in proteins.[10] Another

* Author to whom all correspondence should be addressed.

CCC 0192-8651 / 94 / 101127-12

popular application is the generation of molecular surfaces for graphic display.[11-15]

Depending on the type of study, we may use different types of surface definitions (e.g., equipotential surface, equidensity surface, van der Waals surface). Among them there is a subset that shares a common trait: They consider that a molecule may be represented as a set of rigid interlocking spheres. There are three such surfaces: (1) the van der Waals surface (WSURF), which is the external surface resulting from a set of spheres centered on the atoms or group of atoms forming the molecule (Fig. 1a); (2) the accessible surface (ASURF) (Fig. 1b), defined by Richards and Lee[16] as the surface generated by the center of the solvent, considered as a rigid sphere, when it rolls around the van der Waals surface; and (3) the Solvent-Excluding Surface (ESURF) (Fig. 1c), which was named by Richards[17] as the Molecular Surface and defined by him as composed of two parts: the contact surface and the reentrant surface. The contact surface is the part of the van der Waals surface of each atom which is accessible to a probe sphere of a given radius. The reentrant surface is defined as the inward-facing part of the probe sphere when this is simultaneously in contact with more than one atom. Throughout our experience in this field, we have observed that there is a little confusion about the names of the different types of molecular surfaces. Thus, a typical misunderstanding is to speak about the Accessible Surface when one means the Molecular Surface. We think that part of the problem is that the Molecular Surface (note the capital letters) has a name that reminds one of the generic name of the molecular surface (note the lowercase letters). Because of this, we suggest the name Solvent-Excluding Surface (ESURF), which is more specific than Molecular Surface. We define ESURF as the surface envelope of the volume excluded to the solvent, considered as a rigid sphere (probe sphere), when it rolls around the van der Waals surface. This definition is equivalent to the definition given by Richards, but we think that it is more concise and simple.

Many algorithms have been developed for computing the area and volume of the WSURF and ASURF, but to our knowledge only two of them are able to compute the ESURF.[18-21] GEPOL,[19] developed in our laboratory, computes the three types of surfaces. GEPOL has shown its capability in computing (with a high degree of accuracy and precision) the WSURF and ASURF for all kinds of systems.[22-24] The accuracy in the computation of the ESURF for systems of small or medium size,

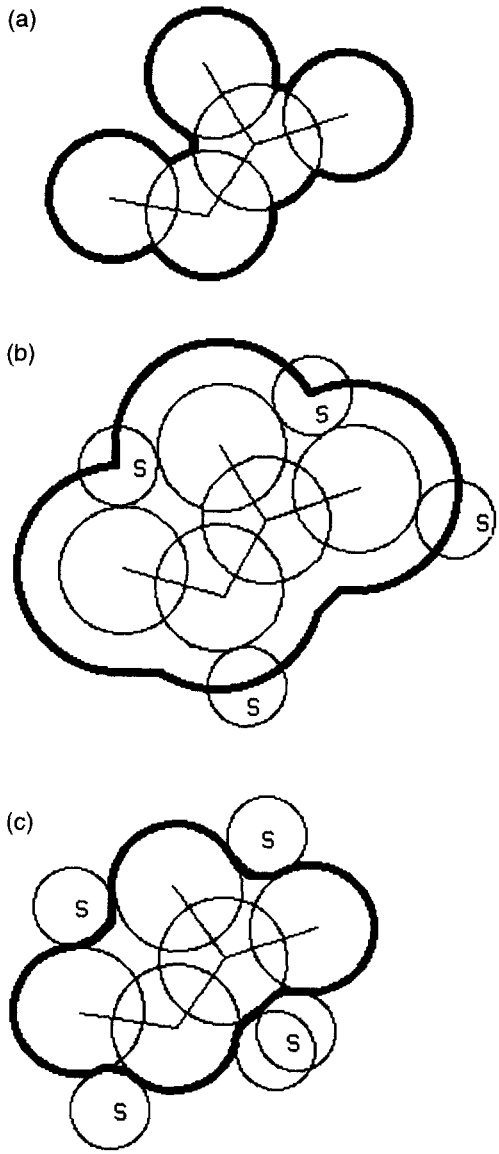


FIGURE 1. (a) van der Waals Surface (WSURF). (b) Accessible Surface (ASURF). (c) Solvent-Excluding Surface (ESURF).

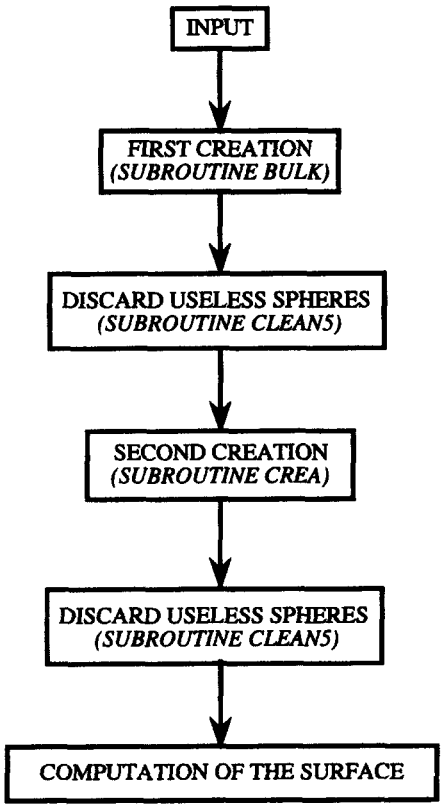
such as organic compounds, has also been shown.[22,23] However, the application of this algorithm to large systems, such as biomacromolecules, presents problems of convergency because the computation of ESURF is so long that, in some cases, it cannot be achieved. In this article we describe the new version of the algorithm of GEPOL (GEPOL93), which (keeping the main idea of the former algorithm) overcomes this problem. The new algorithm has a structure different from the former. The number of parameters used in the computation of the ESURF has been reduced. The

changes introduced for the computation of the ESURF make GEPOL93 not just a new version but a new program. The computations of the WSURF and ASURF remain as in older versions.

Method

As we stated earlier, the new algorithm pursues the same idea as older versions (i.e., it fills the spaces which are not accessible to the solvent with a set of new spheres). Then the algorithm computes the surface for the resulting set of interlocking spheres. This last step is the same as is used to compute the WSURF and ASURF. The changes introduced only affect the creation of the new spheres. We describe briefly the procedure used to compute the surface of a given set of spheres. A more detailed description may be found elsewhere.[24] Once the set of spheres has been defined, their spherical surfaces are divided into 60 spherical triangular tesserae using a data-coded pentakisdodecahedron. The algorithm eliminates all triangles found at the intersection volume of the spheres, taking as reference their centers. Then a fast algorithm permits an efficient calculation of the actual molecular area, volume, and the coordinates of the centers of the remaining tesserae. To increase the accuracy, the initial set of triangles may be divided successively by 4. This process of division is controlled from the input through the parameter NDIV. Therefore, if NDIV is 1, 2, 3, 4, or 5, the number of spherical triangles per sphere will be 60, 240, 960, 3840, and 15,360, respectively.

The changes introduced in GEPOL correspond to the algorithm that creates the new spheres, which will fill the spaces not accessible to the solvent. A diagram showing the main steps of the new procedure for creating new spheres is given in Scheme I. Two steps correspond to the creation of the spheres. The difference between them is the way that the spheres are created. The first creation fills the spaces not accessible to the solvent faster than the second one but produces a less accurate result. The second creation refines the surface. Both steps are followed by a step which discards those spheres created that do not do any work (i.e., those spheres that do not participate in the construction of the ESURF).

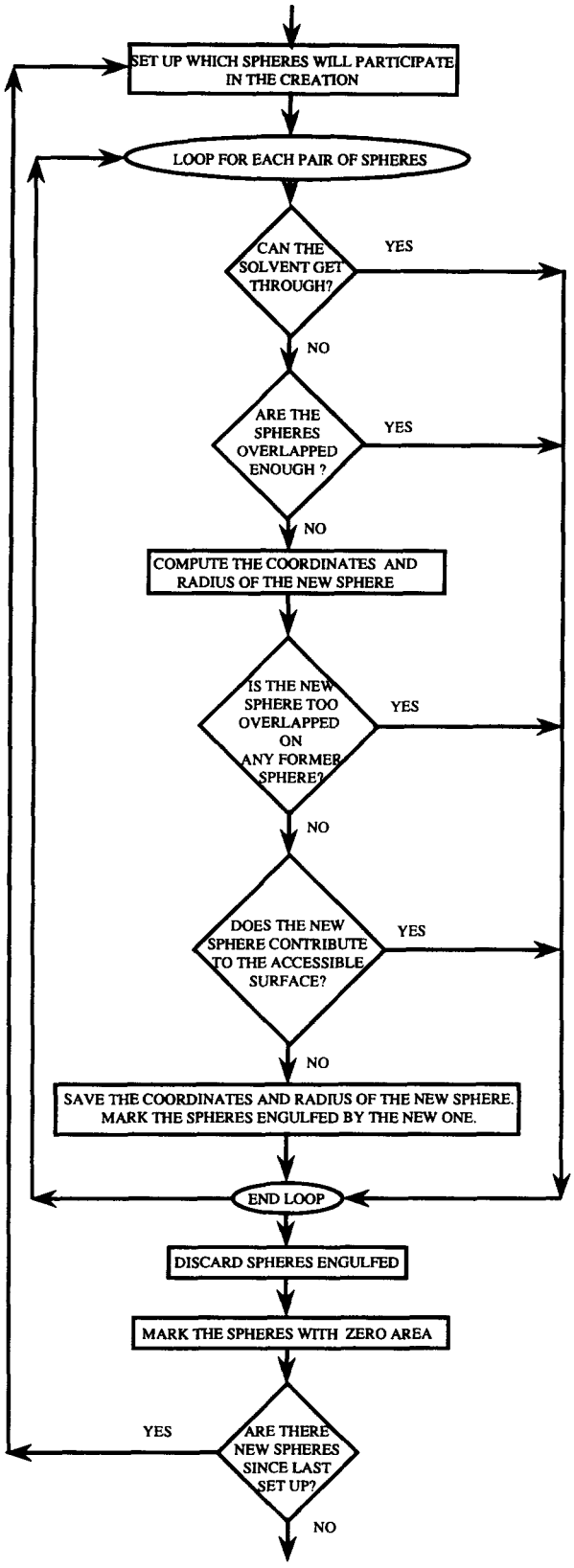


SCHEME 1. General flow diagram of the program.

DESCRIPTION OF SUBROUTINE BULK

Scheme II shows a flow diagram for subroutine BULK. The steps are as follows:

1. First we select the spheres that are going to participate in the creation of new spheres. The first time, they are the spheres given in the input.
2. The procedure analyzes each pair of spheres to decide whether a new sphere should be built between them. Thus a set of questions and actions related to each pair occurs, as shown in Scheme II.
3. Can the solvent get through the pair? If the answer is no, it means that there is a space that is not accessible to the solvent sphere and that a new sphere may be needed. If the answer is yes, the pair will not be analyzed further and a new sphere will not be created. The program is really asking if the distance between the spheres of the pair is larger than $(R_i + R_j + 2R_s)$. R_i and R_j are the radii of the pair of sphere, and R_s is the radius of the solvent sphere.
4. Are the spheres overlapped enough? We define the overlapping distance between two



SCHEME 2. Flow diagram for subroutine BULK.

spheres by the distance p shown in Figure 2. We may define a relative overlapping (p_i^r) with respect to the smaller sphere i by dividing p by the diameter of i. When p_i^r is equal to zero, the spheres are just touching; when p_i^r is one or greater, sphere i is engulfed by sphere j. Rearranging the equation as shown in Figure 2, we have the distance between the spheres in function of p_i^r, R_i, and R_j. p_i^r is one of the parameters that GEPOL93 uses, and we have named it the overlapping factor (OFAC). Thus, the real question is whether the actual distance between the spheres is smaller than the distance needed for them to overlap with a factor equal to OFAC. If they are not overlapped enough, a new sphere may be formed between them. However, if they are, the creation of a new sphere is discarded.

5. The program computes the coordinates and radii of the new candidate sphere. The candidate sphere will have its center at a point lying on the axis that joins the centers of the pair of spheres and equidistant from their surfaces (the formulation is given elsewhere[23]). The radius of the new sphere will be equal to the radius of the largest sphere of the pair.

6. Is the new sphere overlapped too much with any of the spheres already existing? When that occurs, the new sphere is occupying a space that is already occupied to a large extent by another sphere. In such a situation, the presence of the new sphere does not modify appreciably the solvent-excluding surface, and thus the new sphere is discarded. The algorithm works as follows: First, it is found whether the new sphere engulfs or is engulfed by another sphere. If the square of the distance between the centers of spheres i and j is smaller than $(R_i - R_j)^2$, one sphere is engulfed by the other. To know which one is engulfed, the program only needs to determine which is the smaller one. The new sphere is discarded if it is engulfed by another. The spheres engulfed by the new one are marked and are deleted later on. Second, the overlapping is computed, taking into account the OFAC parameter. The relative overlapping is, in this case, calculated with respect to the diameter of the new sphere. If the

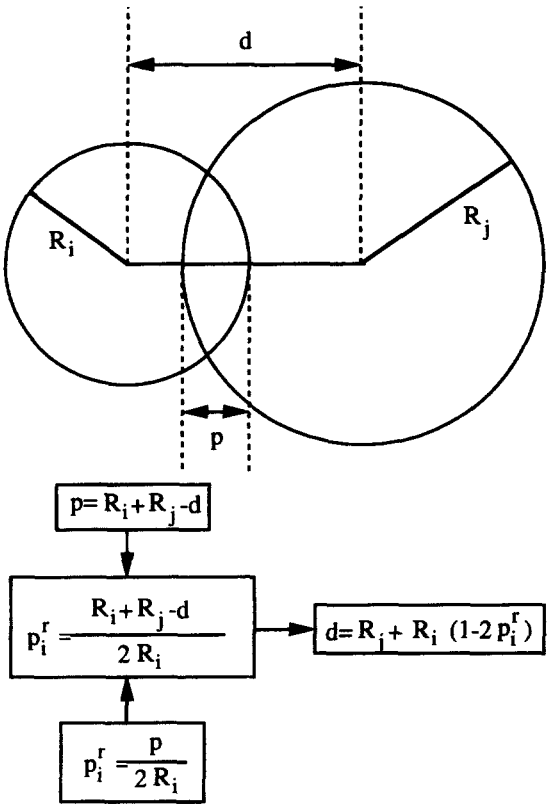


FIGURE 2. Description of the overlap between two spheres (see text).

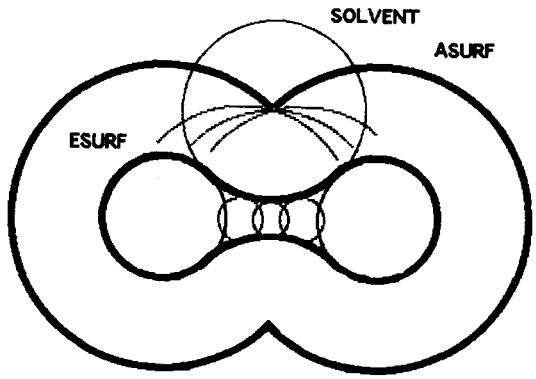


FIGURE 3. Accessible surface and Solvent-Excluding Surface of a system formed by two initial spheres. The set of new spheres (small dashed circumferences) participating in the ESURF do not participate in the ASURF.

new sphere is overlapped too much, it will be discarded.

7. When the radius of the new sphere was determined, we did not take into account the solvent size, so it may happen that its radius is too large. The new sphere with a radius which is too large is discarded. What does too large mean? To clarify this point, Figure 3 shows the ESURF and the ASURF for a system formed initially by two spheres and a solvent of given radii. Three smaller spheres have also been drawn (dotted line) that correspond to the set of new spheres of optimum radii that GEPOL will create to fill the volume not accessible to the solvent. Those spheres have an important characteristic: They do not contribute to the ASURF. In fact, only the initial spheres that form the WSURF can participate in the ASURF. Thus, if we compute the ESURF with GEPOL and later use the full set of original and new spheres to compute the ASURF, the new ones will not contribute to the final result. To discard spheres with a large radius, the program checks for each new sphere if it has at least one point going out of the ASURF formed by the initial set of spheres. If that occurs, the new sphere is discarded. When we work with large systems such as proteins, many of the initial spheres (atoms) are far from the surface, as are the new spheres created among them. In these cases, it is more efficient to create spheres with large radii than many spheres with smaller radii. This subroutine has taken its name from this fact; the spheres are in the bulk of the system.
8. If the candidate to be a new sphere passes the foregoing steps, the program saves its coordinates and radii.
9. The process is repeated from c step 3 for all the possible sphere pairs set up in step 1. Once all of them have been analyzed, the program passes to the next step.
10. All the created spheres that have been engulfed are deleted from the vectors that contain the coordinates and radii.
11. The program finds spheres with zero area. The algorithm determines whether a sphere has zero area by computing a distribution of points on the surface of this sphere. If there is at least one point that is not inside another sphere, the sphere under analysis does not have zero area. As explained earlier, the number of points over the surface is con-

trolled by NDIV. Those spheres with zero area are marked and will not be used in the creation of further spheres.

12. The process is repeated from step 1. The new spheres created, in each of these cycles, are said to belong to the same generation. At the start of each cycle, the pairs are formed with the old spheres and the new ones, excluding those that have been analyzed in a former generation. The algorithm repeats the process until no more spheres are created.

DESCRIPTION OF SUBROUTINE CREA

Scheme III shows a flow diagram of subroutine CREA. As BULK, this subroutine also creates a new set of spheres. However, now the solvent is taken into account when the radii of the new spheres are created. As we said before, CREA refines the set of spheres obtained by BULK. As can be seen, it has many similarities with BULK. So we will only describe the parts that are different. This subroutine creates three types of spheres: A, B, and C. Their characteristics and the expressions for calculating their coordinates and radii are extensively explained in ref. 23.

After checking that the solvent cannot get through the pair, the algorithm checks if the spheres of the pair are overlapped. If that is true, the pair will generate a sphere of type A; otherwise it will be of type B or C. After checking that the pair is of type A, the algorithm checks if the spheres are overlapped enough. This question is solved as in the subroutine BULK. Then the radius of the new sphere is calculated. If the radius is smaller than a parameter named RMIN, the new sphere is discarded. The parameter RMIN is given in the input and can be seen as the radius of the smallest sphere that can be created. If the sphere is big enough, the coordinates of the new sphere are calculated.

If, however, the spheres of the pair are not overlapped, the pair should be of type B or C. The first thing is to check that the sphere that is going to be created is larger than RMIN; if it is not, the pair is discarded. The difference between the pairs of type B and C is that in B the distance between the surface of the solvent (when it is touching both spheres of the pair) and the axis of the pair is larger than the parameter RMIN. In this way, the algorithm identifies the two types of pairs. Following this, the coordinates of the new sphere are calculated.

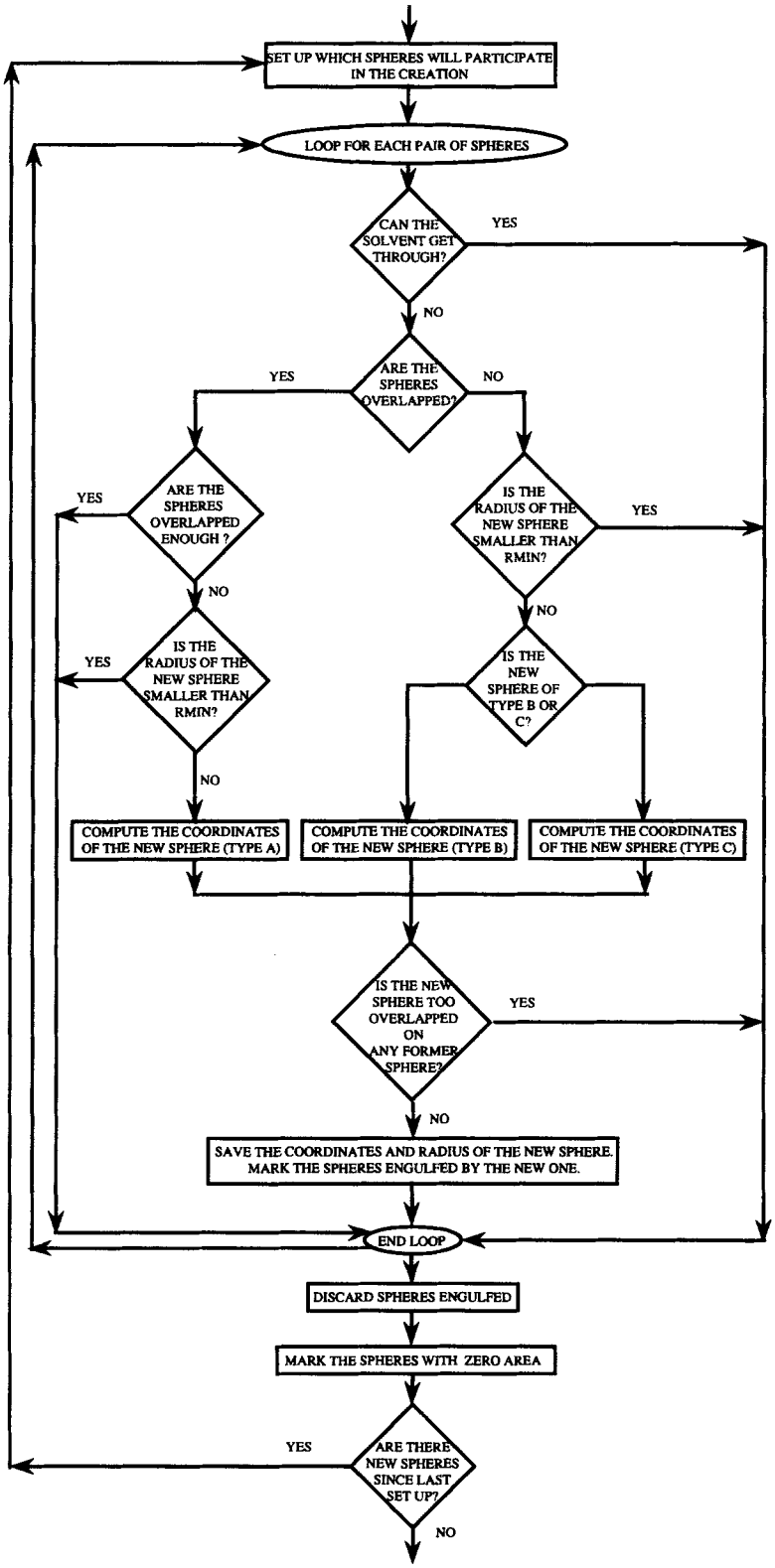
GENERAL ASPECTS OF THE ALGORITHM

To understand the work of the subroutine CLEAN5, we show an arbitrary set of spheres in Figure 4 together with the resulting envelope (thick line). Among the spheres, there are two with zero area (i.e., they do not make a net contribution to the final envelope). However, one of them (dotted line) is essential for a good description of the surface. If we take this sphere out, the resulting envelope will be formed by two surfaces, one exterior and one interior delimiting a cavity. With this example, we show that there are spheres with final area zero that are important in the computation of the surface. However, there are others (such as the one shown with the dashed lines) that have no effect when deleted. The latter are the spheres deleted by subroutine CLEAN5.

As we have seen, GEPOL creates the new set of spheres in an iterative process that could go to infinity if we do not put a brake on. We may classify the brakes that GEPOL uses in two parts: the brakes that the user can control from the input and the intrinsic ones. From the input we may control the creation with two parameters: OFAC and RMIN. As we saw, OFAC tells the program how overlapped the new set of spheres should be; the more overlapped, the closer will be the calculated ESURF to the real one. OFAC may change between 0, spheres not overlapped, and 1, spheres totally overlapped. RMIN is the value of the radius of the smallest sphere that may be created. Thus the smallest RMIN gives the best ESURF. As OFAC tends to 1 and RMIN to 0, the number of new spheres tends to infinity. The intrinsic brakes are the discarding of engulfed spheres; the avoidance of spheres with zero area participating in the process of creation; and the work made by subroutine CLEAN5.

Results and Discussion

To analyze the dependence of the area and volume on the parameters used, we have selected the calcium binding protein from bovine intestine named Calbindin D9k, identified as entry 3ICB at the Brookhaven Protein Data Bank. The X-ray study[25] of 3ICB revealed that its 75 residues form a structure consisting of two helix–loop–helix structures linked by a linker segment. We have centered spheres in all the atoms except hydrogen.



SCHEME 3. Flow diagram for subroutine CREA.

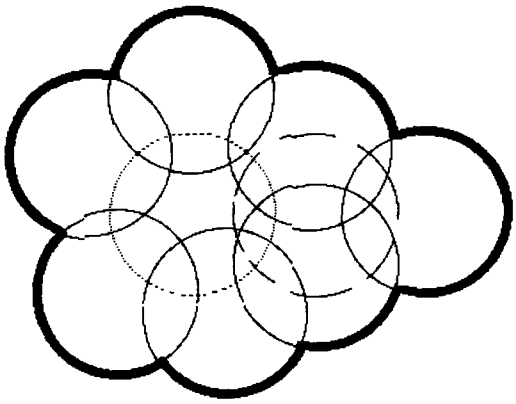


FIGURE 4. Two types of spheres without a net contribution to the final envelope of a given set of spheres. The sphere with a dotted line is essential to obtain a good envelope. The sphere with a dashed line does not do any work.

The radii of the spheres are the same as used by Rose et al.,[26] using 1.4 Å as radius for the solvent. We have not selected this protein for its biological interest but because of its size. It is small enough to allow us to make many computations without an excessive CPU time, and it is big enough to represent the general characteristics of a protein.

BEHAVIOR WITH NDIV

For a given value of OFAC and RMIN, NDIV relates to the accuracy of the method. In a previous article[24] we analyzed the effect of the value of NDIV in computing the area and volume of the WSURF and ASURF. Next we will analyze that effect in the computation of the ESURF. Once the new set of spheres has been created, the algorithm follows the same path to compute the WSURF, ASURF, and ESURF. However, the set of spheres that form ESURF usually has a different characteristic from the set for WSURF and ASURF: The overlapping among the spheres in ESURF is higher than in the other two types. Thus many of the spheres will contribute to the surface of the molecule with a small piece. If the density of points over the surface is too low, many of these pieces will get lost and the final area will be underestimated.

Figure 5 shows the values of the area and volume of the ESURF obtained with different values of OFAC and NDIV. The points corresponding to values obtained with the same NDIV are joined by a line. The value of RMIN was set to 0.5 for all these computations. In general, for the five curves

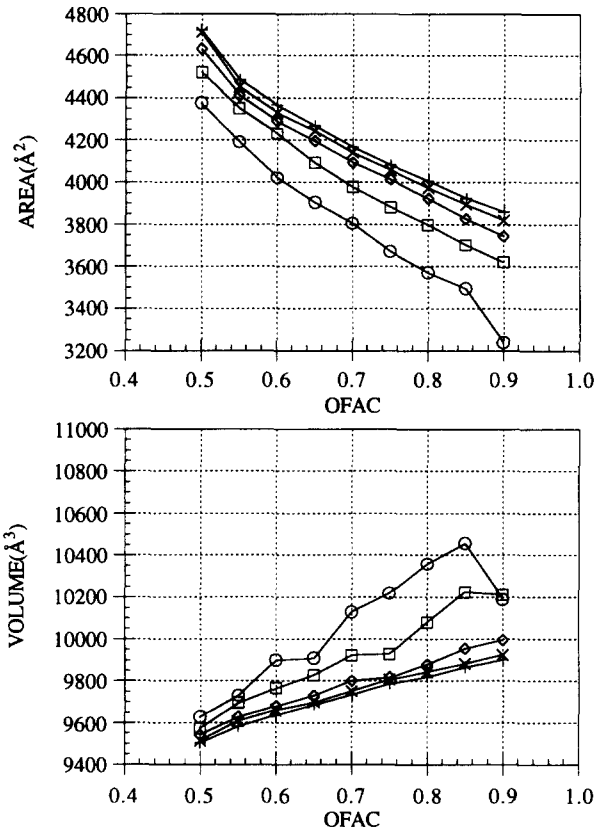


FIGURE 5. Area and volume of the Solvent-Excluding Surface versus the overlapping factor (OFAC). Each curve was calculated with RMIN = 0.5 and with a different value of NDIV: 1(○), 2(□), 3(◇), 4(×), 5(+).

the area decreases and the volume increases as the overlapping factor among the spheres increases. The starting point of the computation is the set of spheres forming the WSURF; this surface has a larger area and smaller volume than the ESURF (see Fig. 1). Thus the closer the computed ESURF is to the real ESURF, the smaller will be the value of its area and the higher the volume enclosed. Therefore, the tendency is the expected one.

As can be seen in Figure 5, the differences of the area and volume obtained with correlative values of NDIV decrease when NDIV increases, in a convergent trend. This result is coherent with the fact that the number of triangles (points) used to describe each sphere increases with NDIV. Figure 6 shows for each value of OFAC the relative difference between the values obtained with NDIV 5 and NDIV 1, 2, 3, and 4. Again the convergency increasing NDIV is clear. The relative differences among the values of NDIV 4 and 5 are not larger than 1% for the area and 0.3% for the volume.

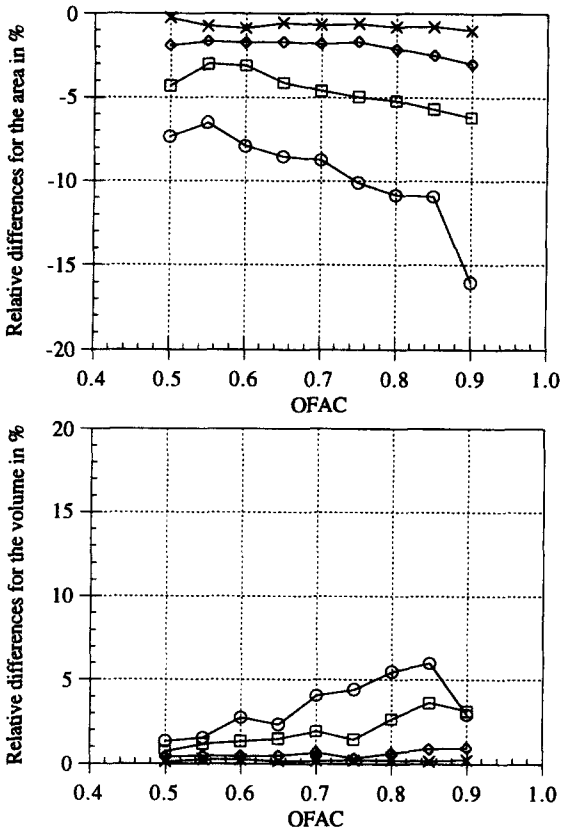


FIGURE 6. Relative differences between the areas and volumes obtained with NDIV 1(○), 2(□), 3(◇), and 4(×), and the values obtained with NDIV = 5 in %, versus OFAC. With RMIN = 0.5.

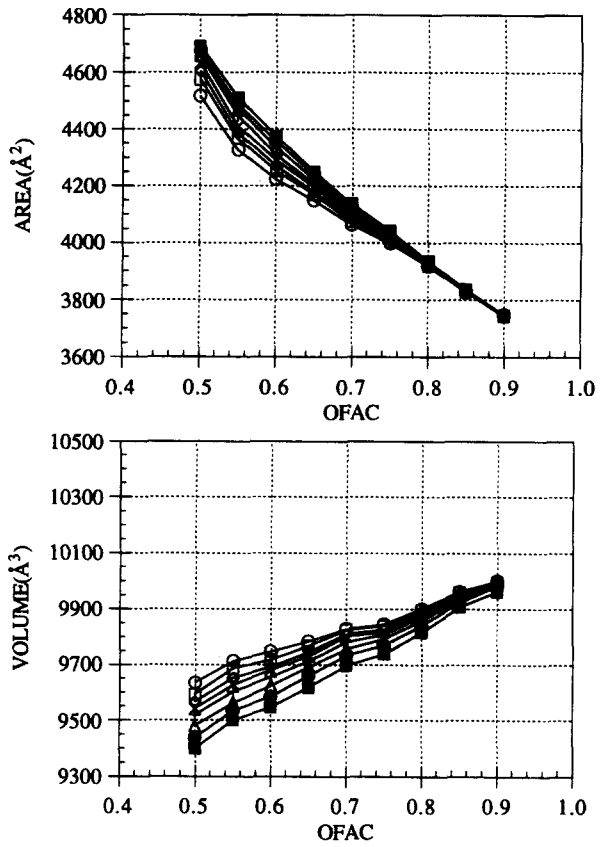


FIGURE 7. Area and volume of the Solvent-Excluding Surface versus the overlapping factor (OFAC). Each curve was calculated with NDIV = 3 and a different value of RMIN: 0.2(○), 0.3(□), 0.4(◇), 0.5(×), 0.6(+). 0.7(△), 0.8(●), 0.9(■).

Taking into account these last two facts, we may accept that values obtained with NDIV 5 are so close to the limiting value that they may be taken as exact for the computation of the errors in the area and volume when another value of NDIV is used. If we accept this hypothesis, Figure 6 may be seen as a representation of the relative errors when using NDIV 1, 2, 3, and 4. As can be seen, the relative error in calculation of the volume is lower than for the area. In addition, as the overlapping factor increases, the errors at some point start to increase. As the overlapping among the spheres increases, the number of spheres that participate in the surface with a small piece increases. To describe these smaller pieces properly, a higher density of points over the surface is needed. For that reason, it is best to use a high value of NDIV with a high value of OFAC. Thus, for NDIV 1, 2, 3, and 4, this fact starts to be present for values of OFAC larger than 0.55, 0.6, 0.8, and 0.85, respectively, as may be seen in Figure 6.

BEHAVIOR OF OFAC AND RMIN

Figure 7 shows the values of the area and volume for several values of OFAC and RMIN obtained with NDIV = 3. The points with same value of RMIN have been joined with a line. As in Figure 5, the area decreases and the volume increases as OFAC increases. A similar trend of the area and volume is observed with RMIN. However, the differences among the values obtained with different RMIN are smaller as OFAC increases. Thus for values of OFAC higher than 0.8, the use of a specific value of RMIN has very little effect on the total value of the area. Even though this trend also exists for the volume, it is more sensitive to the value of RMIN selected.

CPU TIME

From Figures 1 through 7, we see clearly that as NDIV and OFAC increase and RMIN decreases,

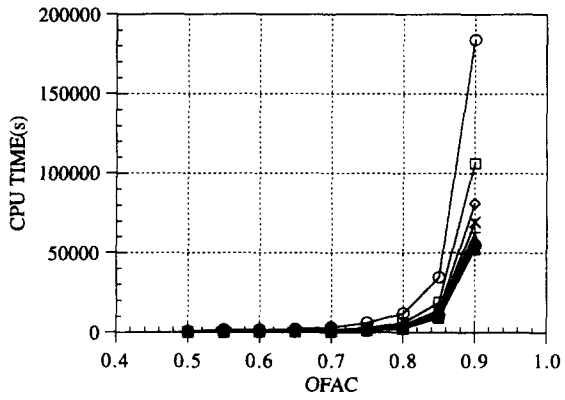


FIGURE 8. CPU time expended in the computation of the Solvent-Excluding Surface versus OFAC. Each curve was obtained with NDIV = 3 and a different value of RMIN: 0.2(○), 0.3(□), 0.4(◇), 0.5(×), 0.6(+), 0.7(△), 0.8(●), 0.9(■).

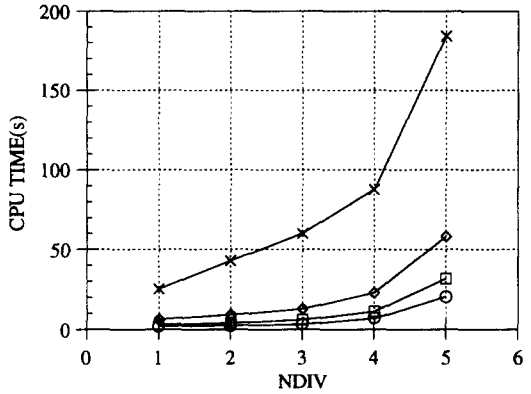


FIGURE 9. CPU time expended in the computation of the Solvent-Excluding Surface versus NDIV. Each curve was obtained with RMIN = 0.5 and a different value of OFAC: 0.5(○), 0.6(□), 0.7(◇), 0.8(×).

the results obtained are better. Thus, why not use higher values of NDIV and OFAC and lower values of RMIN? Figure 8 shows the CPU time employed to compute the surface for several values of OFAC and RMIN, using in all of them NDIV = 3. Values obtained with the same RMIN are joined with a line. The computations were carried out in a VAX 4000-200, so the CPU time corresponds to a machine of its characteristics. The time increases exponentially as OFAC increases, tending to infinity as OFAC tends to 1. That is the result to be expected if we think of OFAC as the overlapping factor. Thus, more and more spheres will be created as OFAC increases to fulfill the condition of overlapping. The CPU time behaves in a way similar to RMIN. Figure 9 shows a similar graph but for several values of NDIV and OFAC, using in all of them RMIN = 0.5 Points with equal OFAC are joined with a line. The variation of the CPU time with NDIV is also important. But in the range of our work, NDIV between 1 and 5 is not so critical as in the previous parameters.

ESTIMATION OF THE ERROR

Determining the exact area and volume for systems with many spheres is impossible. However, if we accept that the limiting value of the area and volume obtained with GEPOL93 when OFAC tends to 1, RMIN to 0.0, and NDIV to infinity is the exact value, we will be able to estimate the error when using other values for the parameter.

If we observe Figure 5, we will see that for NDIV = 5 the relation of the area with the higher values of OFAC is linear. Thus, to compute the exact value, we fit the four last points of that curve to a line and we extrapolate up to OFAC = 1. The extrapolated value is 3705 Å^2, and the correlation factor of the regression is .9991. A similar computation was made with the volume, with the exact value of 9980 Å^3 and the correlation factor .998. In doing this regression, we have assumed the following points: (1) Area and volume change linearly with OFAC for its high values; (2) for high values of OFAC, the total area and volume depend very little on RMIN (thus, the curve used for the fit was calculated with RMIN = 0.5); (3) the value obtained with NDIV = 5 is very close to the limiting value (i.e., the value obtained when NDIV tends to infinity). Using the extrapolated values of the area and volume as the exact ones, we have calculated the relative errors for several values obtained with NDIV = 3 and different values for RMIN and OFAC (Fig. 10). The errors are lower in the volume than in the area.

To check if this kind of estimation of the error is valid for other molecules, we repeated a similar analysis for a set of systems of different sizes. Thus, we studied the following systems: two spheres of 2 Å of radius separated 5 Å (TWO); the residue number 4 of the 3ICB,[25] which is a glutamate (GLU); the monomer of the melittin (MLT)[27]; and the molecule of Lysozyme (LYSO).[28] We computed, for all of them, the area and volume using NDIV = 5, RMIN = 0.5, and four different values of OFAC: 0.75, 0.80, 0.85, 0.90. Then we fitted the values of the area and volume to a straight line and obtained the extrapolated value for OFAC = 1. Taking those values as the exact ones, we com-

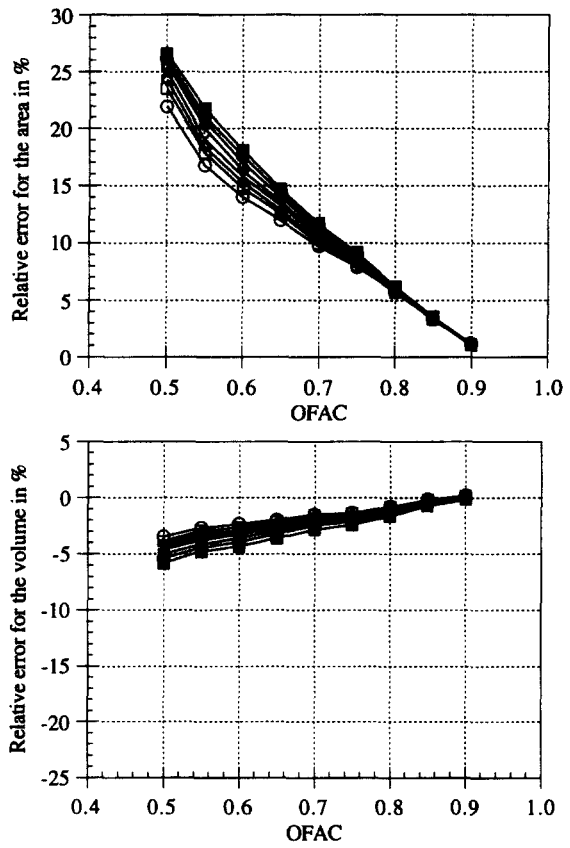


FIGURE 10. Estimated relative error in the computation of the area and volume of the Solvent-Excluding Surface versus OFAC. Each curve was obtained with NDIV = 3 and different value of RMIN: 0.2(○), 0.3(□), 0.4(◇), 0.5(×), 0.6(+), 0.7(△), 0.8(●), 0.9(■).

puted the estimated error for all the structures when using NDIV = 3, OFAC = 0.8, and RMIN = 0.5. For the case of the two spheres, the true exact value and the true errors can be calculated from classical mathematics. Table I shows the estimated errors obtained for the area and volume in these structures. In the second column, the number of atoms with an initial sphere is given. The errors are given with the appropriate sign. Thus, as we have seen, the area is overestimated and the volume underestimated. The absolute values of the errors are smaller for the volume than for the area. The values for the different structures are similar. This result shows the coherence in the method used to estimate the errors. The estimated errors in the area are smaller for the smaller systems than for the larger ones. We think that this difference is because for small systems the ESURF is just a smoothing of the WSURF, given its simplicity. However, in large systems the ESURF is notably different from the WSURF and the convergence with the parameters of GEPOL may be slightly slower. As default values, the program uses NDIV = 3, OFAC = 0.8, and RMIN = 0.5. This set of parameters, as we have shown, gives a good enough result. However, if a better result is desired, NDIV and OFAC may be increased and RMIN decreased.

The estimated errors should not been taken as true errors given that they have been obtained under several hypotheses. To check the acceptability of these hypotheses, we have computed the exact value of the area and volume for the case of two spheres. Only in this case can we calculate the exact values, using classical mathematics. Using these values, we have calculated the error for the values obtained with GEPOL (NDIV = 3, OFAC = 0.8, RMIN = 0.5): The errors are +0.52 and −0.16 for the area and volume, respectively. They are lower than the estimated ones. This difference may be produced by use of the extrapolated values in calculating the estimated errors. Thus, we have computed the area and volume for the two smaller systems with values of OFAC of 0.95 and 0.98. In both cases, we observed that the slope corresponding to the variation of the area (and volume) with OFAC decreases when OFAC approaches to 1. All these facts suggest that the true errors may be smaller than the estimated ones.

Another way to estimate if our method works properly is to compare the area obtained with GEPOL93 with the other existing algorithms. To our knowledge, there is only one other method, named MSDOT,[18] that is widely used to compute the ESURF. We have computed the total ESURF area using both methods for the same systems used in Table I. For the computations with GEPOL93, we have used the default parameters given in the program (NDIV = 3, OFAC = 0.8, and RMIN = 0.5). For the computation with

TABLE I.
Estimated Errors of the Area and Volume for Several Molecules when Using NDIV = 3, OFAC = 0.8, and RMIN = 0.5.

Molecule	Number of atoms	Estimated errors in %	
		Area	Volume
TWO	2	+1.3	−0.73
GLU	9	+3.8	−1.6
MLT	200	+5.6	−1.0
3ICB	602	+5.8	−1.0
LYSO	999	+6.4	−0.67

MSDOT, a density of 20 dots/Å^2 has been used (this value is the highest density that can be used by MSDOT with spheres of 2 Å radius). The differences between the areas calculated by both methods are around ±3%. Given that both methods have algorithms which are so different, these similar results suggest that we may trust the value of the total area obtained with them. However, as we describe in previous articles[22,23] GEPOL is more precise and describes better small parts of the surface and small changes in the surface than MSDOT, which has an oscillating[29] behavior in those cases.

Conclusions

We have presented the new GEPOL93 program that computes the Solvent-Excluding Surface. GEPOL93 follows the same concept as former versions of GEPOL, but with a full new algorithm. Thus, it computes the Solvent-Excluding Surface by filling the spaces not accessible to the solvent with a set of new spheres. The computation is controlled by three parameters: the number of triangles per sphere, controlled by NDIV; the maximum overlap among the new spheres (OFAC); and the size of the smallest sphere that can be created (RMIN). NDIV may take integer values between 1 and 5, OFAC real numbers between 0 and 1, and RMIN a real positive number. We have analyzed the behavior of the method with the three parameters. Thus, the result improves as NDIV and OFAC increase and RMIN decreases. In addition, we have estimated the error for the area and volume obtained in function of the three parameters.

Acknowledgments

We thank Mr. Charles Morton for his great help in preparing the English version of all our articles and for his decisive help in redefining the Molecular Surface as the Solvent-Excluding Surface. We thank Dr. R. M. Jackson for his helpful comments about the problems that former versions of GEPOL have with proteins. I. T. thanks the Ministerio de Educacion y Ciencia (MEC) (Spain) for a grant. This work was supported by DGICYT (project PS90-0264 and PB93-0699).

References

1. J. L. Pascual-Ahuir, E. Silla, J. Tomasi, and R. Bonaccorsi, *J. Comp. Chem.*, **8,** 778 (1987).
2. F. M. Floris, J. Tomasi, and J. L. Pascual-Ahuir, *J. Comp. Chem.*, **12,** 784 (1991).
3. V. Dillet, D. Rinaldi, J. G. Ángyán, and J. L. Rivail, *Chem. Phys. Lett.*, **202,** 18 (1993).
4. C. J. Cramer and D. G. Truhlar, *Science*, **256,** 213 (1992).
5. I. Tuñon, E. Silla, and J. L. Pascual-Ahuir, *Protein Eng.*, **5,** 715 (1992).
6. I. Tuñon, E. Silla, and J. L. Pascual-Ahuir, *Chem. Phys. Lett.*, **203,** 289 (1993).
7. R. B. Herman, *J. Phys. Chem.*, **76,** 2754 (1972).
8. K. A. Sharp, A. Nicholls, R. F. Fine, and B. Honig, *Science*, **252,** 106 (1991).
9. A. E. Eriksson, W. A. Baase, H.-J. Zhang, D. W. Heinz, M. Blaber, E. P. Baldwin, and B. W. Matthews, *Science*, **255,** 178 (1992).
10. J. L. Pascual-Ahuir, E. L. Mehler, and H. Weinstein, *Mol. Eng.*, **1,** 231 (1991).
11. L. H. Pearl and A. Honegger, *J. Mol. Graphics*, **1,** 9 (1983).
12. R. Voorintholt, M. T. Kosters, G. Vegter, G. Vriend, and W. G. J. Hol, *J. Mol. Graphics*, **7,** 243 (1989).
13. J. B. Moon and W. J. Howe, *J. Mol. Graphics*, **7,** 109 (1989).
14. W. Heiden, M. Schlenkrich, and J. Brickmann, *J. Comp. Aided Mol. Design*, **4,** 255 (1990).
15. E. Silla, F. Villar, O. Nilsson, J. L. Pascual-Ahuir, and O. Tapia, *J. Mol. Graphics*, **8,** 168 (1990).
16. B. Lee and F. M. Richards, *J. Mol. Biol.*, **55,** 379 (1971).
17. F. M. Richards, *Ann. Rev. Biophys. Bioeng.*, **6,** 151 (1977).
18. M. L. Connolly, QCPE Program No. 75 (1981).
19. J. L. Pascual-Ahuir, E. Silla, and I. Tuñon, QCPE Program No. 554 (1988, 1992, 1994).
20. R. J. Zauhar and R. S. Morgan, *J. Comp. Chem.*, **11,** 603 (1990).
21. J. Greer and B. Bush, *Proc. Natl. Acad. Sci. USA*, **75,** 303 (1978).
22. J. L. Pascual-Ahuir and E. Silla, In R. Carbó, Ed., *Studies in Physical and Theoretical Chemistry*, Elsevier Science B.U., Amsterdam, 1989, p. 597.
23. J. L. Pascual-Ahuir and E. Silla, *J. Comp. Chem.*, **11,** 1047 (1990).
24. E. Silla, I. Tuñon, and J. L. Pascual-Ahuir, *J. Comp. Chem.*, **12,** 1077 (1991).
25. D. M. E. Szebenyi and K. Moffai, *J. Biol. Chem.*, **261,** 8761 (1986).
26. G. D. Rose, A. R. Geselowitz, G. J. Lesser, R. H. Lee, and M. H. Zehfus, *Science*, **229,** 834 (1985).
27. D. Eisenberg, M. Griskov, and T. C. Terwilliger, *Brookhaven Protein Data Bank*, Entry 2mlt.ent (1990).
28. S. Sheriff and D. R. Davis, *Brookhaven Protein Data Bank*, Entry 2hfl.ent (1987).
29. A. Meyer, *Chem. Soc. Rev.*, **15,** 449 (1986).