1981-1 A

AMBER: Assisted Model Building with Energy Refinement. A General Program for Modeling Molecules and Their Interactions

Paul K. Weiner and Peter A. Kollman

School of Pharmacy, Department of Pharmaceutical Chemistry, University of California at San Francisco, San Francisco, California 94143

Received 4 June 1980; accepted 14 January 1981

We describe a computer program we have been developing to build models of molecules and calculate their interactions using empirical energy approaches. The program is sufficiently flexible and general to allow modeling of small molecules, as well as polymers. As an illustration, we present applications of the program to study the conformation of actinomycin D. In particular, we study the rotational isomerism about the D-Val-, L-Pro, and L-Pro-Sar amide bonds as well as comparing the energy and structure of the Sobell model and the x-ray structure of actinomycin D.

INTRODUCTION

One of the goals of the chemist is to explain what happens on a molecular level during a chemical reaction. One approach to this goal is to develop a mathematical model that gives a function value, the potential energy, for any given configuration of the nuclei.

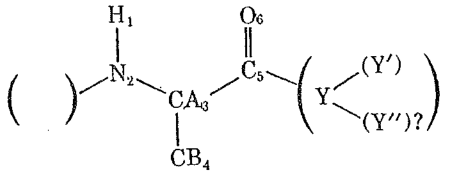
The most rigorous quantum mechanical models, *ab initio* as well as semiempirical, have greatly increased in power over the last decade due to more sophisticated algorithms, as well as decreasing computer costs. However, many problems of biological interest, such as the study of nucleotide and protein conformation, still require the most elementary model empirical energy functions. Though the functions are crude, this approach has been applied successfully to the study of hydrocarbons,[1] peptides and amino acids,[2] and nucleotides[3] during the past few years.

It is interesting that almost every group using empirical energy functions has their own computer program. This is in marked contrast to the use of a few well-documented quantum mechanical programs. The likely reasons for these are the following: (1) the greater difficulty and complexity of the quantum mechanical approaches means many more man years of effort in program development; (2) the lack of well-defined criteria as to what functions are necessary for quantitatively satisfactory molecular force fields means that any terms that might cause programming or database problems are omitted. People can write programs with any desired degree of complexity.

Unfortunately, the user of a hastily designed program soon discovers his limitations. Often, either the input is complex and time consuming or the program lacks generality. Flaws often appear in even carefully designed programs after they have been used extensively. It is one of the authors' (PKW) experience with a carefully written second-generation empirical energy program and its limitations that leads to the writing of a third-generation molecular modeling program at UCSF called AMBER (assisted model building and energy refinement).

The program was developed after a study of several other programs generally available. There are three programs available from the *Quantum Chemistry Program Exchange:* two of these focus on small molecules (MMI[4] and QCFF/PI[5]) and the third on polypeptide conformations, UNICEPP.[6] Another package, CAMSEQ[7] and CAMSEQ/M,[8] has been applied mainly to small molecules of biological interest. Other programs which concentrate

CCC 0192-8651/81/030287-17$03.40



Bonds	Angles	Dihedrals	Excluded atoms
1—2	1—2—3	1—2—3—4	1—2,3,4,5
2—3	2—3—4	1—2—3—5	2—3,4,5,6,Y
3—4	4—3—5	2—3—5—6	3—4,5,6,Y,Y′. . .
3—5	2—3—5	2—3—5—Y	4—5,6,Y,Y′. . .
5—6	3—5—6	4—3—5—6	5—6,Y. . .
5—Y	3—5—Y	4—3—5—Y	6—Y,Y′. . .
	6—5—Y	3—5—Y—Y′(?)	
	5—Y—Y′(?)	3—5—Y—Y″(?)	
	5—Y—Y″(?)		

HB: 1–6 with future or past residues?

Figure 1. Information needed for energy function.

on polymer conformation are REFINE[9] and the Gelin–Karplus[10] program. Our program is closest in spirit to the latter program. Though the implementation differs greatly, the philosophy of modular design has been retained and the same energy functions are used.

The result of the studies is a series of modular programs that handle large and small molecules of any type—protein, nucleotide, or "other"—with a common database. The molecules are divided into pieces of arbitrary size, called residues, which are automatically linked together to form a molecular system. The system may then be edited, minimized, analyzed, and manipulated graphically.

BASIC ASSUMPTION OF EMPIRICAL ENERGY APPROACHES

The basic assumption of empirical energy approaches is that one can replace a Born–Oppenheimer energy surface for a molecule or system of molecules by an analytical function. The potential energy function chosen is generally given as a sum of strain energies and nonbonded interaction terms:

$$E_{\text{total}} = \sum_{\text{bonds}} K_r(r - r_{\text{eq}})^2 + \sum_{\text{angles}} K_\vartheta(\vartheta - \vartheta_{\text{eq}})^2 + \sum_{\text{dihedrals}} \tfrac{1}{2}K\,[1 + \cos(n\varphi - \gamma)] + \sum_{i<j} \frac{B_{ij}}{R_{ij}^{12}} - \frac{A_{ij}}{R_{ij}^{6}} + \frac{q_i q_j}{\epsilon R_{ij}}$$

The first three terms represent the difference in energy between a geometry in which the bond lengths, bond angles, and dihedral angles have ideal values and the actual geometry. The remaining terms represent nonbonded van der Waals (vdW) and electrostatic interactions. Although one could (and often does) use more complex functions to describe bond stretching and bending or different forms of the nonbonded repulsions, the current version of AMBER uses the above functional form, plus a 10–12 H-bond function[10] if so chosen by the user. This decision was based on a lack of parameters for systems other than hydrocarbons.[4]

For a small molecule, one could define all the quantities needed to evaluate this function by hand. This process is illustrated for a small portion of a peptide chain in Figure 1. Even for a six-atom residue, the various quantities needed are numerous. In addition, there is the problem of defining all variables when the form of the next res-

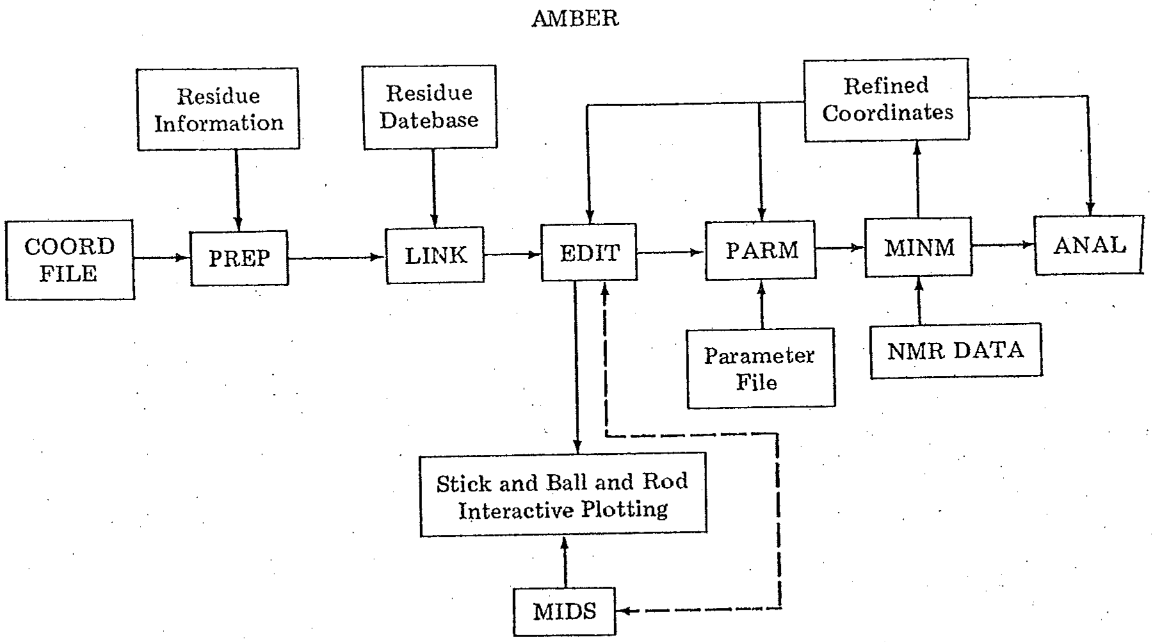


Figure 2. Flow chart of AMBER.

idue is not known. After constructing a few molecules, it becomes obvious that automatic calculation of these quantities and linking of the residues is a very desirable feature.

The major impetus for writing AMBER is to simplify the process of model building and of carrying out such empirical energy calculations for any molecular system.

DESCRIPTION OF PROGRAM

AMBER has been written and developed on the UCSF Computer Graphics Laboratory PDP 11/70. There is also an IBM 360/370 version and a CDC 7600 version developed at LBL. An overall view of the program is given in the flow chart in Figure 2 and a brief summary of the functions of the individual units are given in Table I. Data is passed from one unit to another through binary files written to the disk. These files are read and written with the same structure in all programs with the exception of the PREP output file for the LINK program and the PARM output file for the MINM program. This feature facilitates communication between programs. If the communication is across separate computers, there are options to pass data in formatted rather than binary files. Thus, the input could be prepared on a minicomputer or graphics screen and then sent to a large computer for minimization.

The program is built around the concept of representing the molecule as a tree structure. This is a concept that has been used for many years[11] and is one that is essential for convenient internal coordinate manipulation and automatic linkage

Table I. Functions of sections of AMBER.

PREP	Prepares a single residue
LINK	Links residues together
EDIT	Modifies structure Changes charges Adds counterions Reads in x-ray coordinates
PARM	Adds parameters Prepares input for MINM Flags structural input without PARM
MINM	Minimizes energy partitioning
ANAL	rms Comparison of structures Sugar pucker calculated

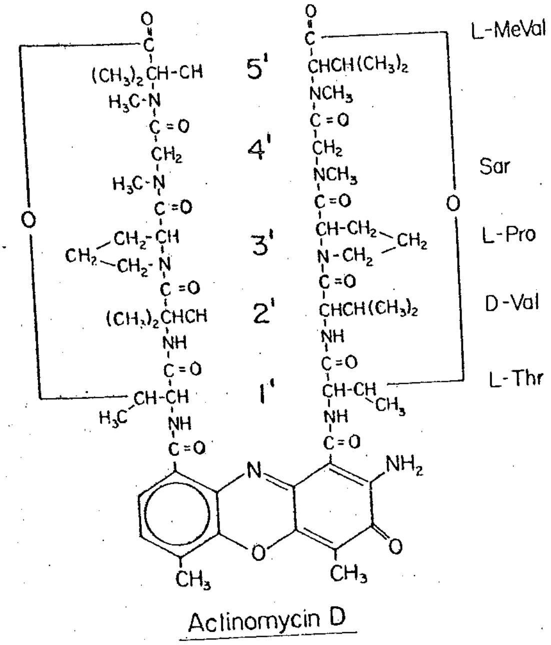


Figure 3. Structure of actinomycin D.

```
1. 1 proline
2. LPRO INT
3.    1 3 1
4. CORRECT OMIT DU BEG CHAR
5. 0.2
6.  7 8
7.
8. 7 8 4 9
9.
10. 1 DUMM DU M 0 0 0 0. 0. 0. 0.
    2 DUMM DU M 0 0 0 1. 449 0. 0. 0.
    3 DUMM DU M 0 0 0 1. 522 111.1 0. 0.
    4 N N M 0 0 0 1. 337 117.6 180. 0.
    5 CD C2 S 0 0 0 1. 458 126.1 356.1 0.
    6 CG C2 S 0 0 0 1. 50 103.2 200.1 0.
    7 CB C2 E 0 0 0 1. 51 106.0 338.3 0.
    8 CA CH M 0 0 0 1. 451 120.6 175.2 0.
    9 C C M 0 0 0 1. 522 111.1 180. 0.
    10 0 0 E 0 0 0 1. 229 120.5 0. 0.
21. −.313 .112 .026 .028 .132 .305 −.290
```

Figure 4. Sample input for PREP.

of segments. The basic units of the tree are as follows:

(1) M—main chain atom. These atoms represent the trunk of the tree and correspond to the backbone of a protein.

(2) S—side chain atom. These atoms represent the straight segments of the branches of the tree. They are connected to 2 other atoms.

(3) B—branch atoms. These atoms form branch points along the branches of the tree. They are connected to three other atoms. If an atom is connected to more than three other atoms, other special symbols are used, such as '3' for a three-way branch.

(4) E—end atoms. These are atoms that are at the ends of the branches.

The alert reader is aware that very few molecules are actually tree structures since loop-closing bonds frequently occur. However, these extra bonds can be identified and their inclusion in the database detracts little from the usefulness of tree structures.

The structure of the programs and its capabilities will now be described. To make the description less abstract, the process of constructing and minimizing actinomycin D will be followed in detail. It is very important that an example be given from start to finish so the reader can judge for himself the convenience of the system.

Actinomycin D is a molecule of biological interest because it binds to double helical DNA and inhibits RNA synthesis. Its structure has been solved crystallographically by cocrystallizing it with deoxyguanosine[12] and its chemical diagram is given in Figure 3. There are two cyclic polypeptide rings with a connecting chromophore. To assemble actinomycin, AMBER only requires the specification of the six unique residues in the PREP program.

PREP

PREP is designed to provide a convenient means of building residues that later can be linked together to form the molecular system. The input consists of Cartesian coordinates, internal coordinates, or both. It is also necessary to specify an unique name (for user identification), a symbol representing atom type (for parameter identification), charge, and a tree structure symbol for each atom.

The input for L-proline, one of the residues of actinomycin D, is given in Figure 4 and described in Appendix A. Here only internal coordinates are used. The special symbol DU represents a dummy atom. These are atoms that help specify the geometry but are not used in the energy calculation. The usefulness of initial dummy atoms will be discussed in the LINK section. After being used, all dummy atoms may be edited out either at the end of PREP or in EDIT.

By default, the program assumes that the hierarchy E, B or S, and M defines the given bonds, bond angles, and torsion angles. In the case of two equivalent symbols on a given atom, the symbol on the atom which has been defined first is given precedence. If a different set of three atoms is specified, the residue is built as desired; then the internal coordinates corresponding to the tree structure are calculated and replace the initial values.

The tree structure may be used to determine all bonds except loop-closing ones. These may either be read explicitly or calculated by specifying a minimum distance below which any two atoms not connected on the tree are assumed to be bonded. Once the bond array is calculated and sorted, it is used to calculate all bond angles and dihedral angles. The use of a sorted list improves the efficiency of this procedure.

Next the bond, angle, and dihedral arrays are used to form a list of excluded atoms (atoms excluded from special nonbonded energy terms). Atoms that form a bond or an angle implicitly have had nonbonded interactions counted through the assumption of ideal bond lengths and angles. Dihedral atoms are included in this list because the end atoms are used in a separate evaluation of nonbonded terms where special weights can be assigned to the interactions. This is particularly useful when minimizing only dihedral angles since the reduced interactions compensate for the rigid geometry of the intervening atoms. PREP flags dihedral angles that occur in cases such as five-membered rings so that no extraneous nonbonded terms will be calculated.

The tree structure also enables the program to calculate what atoms are affected by the rotation about a given bond. Rotations on the graphics screen and the automatic determination of dependent functions both require this information.

Finally a draw routine is present that generates efficient plot commands by following the tree along shorter branches and then along longer ones, while inserting loop-closing bonds at the appropriate places. A concise sequence of plot commands is important in interactive computer graphics where a structure is drawn thousands of times as the user manipulates the molecule.

PREP has options to output a database (for example, a collection of nucleotide and protein residues) as FORTRAN subroutines (the data is converted to alphanumeric format and placed into data statements in a subroutine that the program outputs) for use in an overlay LINK program (IBM and CDC versions) or a series of binary files for a file-oriented LINK program (UNIX version). Individual residue information is output as a formatted file.

LINK

After repeating the above process for the six residues of actinomycin D, the next step is the combination of the residues by LINK. LINK requires the system be divided into discrete segments called molecules, which may correspond to an actual molecule. Residues are the basic unit of each molecule.

Every molecule is identified as P (protein), N (nucleotide), or O (other). If P or N is specified, the program branches to one of two subroutines that refer to a database of common amino acids or nucleotides. The user does not have to refer continually to the files where these database residues have been created. If the residue name is not located in the database, a search is made through the files identified at the beginning of the LINK input.

Residues within a given molecule are identified by a four-letter code. These residues are consecutively covalently linked end M atom to initial M atom of the following residue. As the residues are connected, all array elements discussed in the PREP program are evaluated for the linking atoms.

```
actinomycin
ACTC   0/u/pak/actc.out
THRP   0/u/pak/thrp.out
DVAL   0/u/pak/dval.out
LPRO   0/u.pak/lpro.out
SAR   0/u/pak/sar.out
NMVA   0/u/pak/nmva.out

DU
  21 0 0 0 0.0 0
  actinomycin 1st peptide
0  1
THRPDVALLPROSARCNMVA

     1 50G1 C

actinomycin chromophore
0  1        1  3
ACTC

     1   1N   CO1      1

actinomycin second peptide
0  1        1  3
THRPDVALLPROSARCNMVA

     1   1C09 N      2
     1   50G1 C

QUIT
```

Figure 5. Sample input for LINK.

The rotation array is updated for all M atoms at the completion of the input for each molecule.

The first residue of each molecule after the first may be attached either covalently or noncovalently to any atom of any previous molecule. By attaching the first M atom of the second and later molecules noncovalently to a beginning dummy atom of molecule 1, a common internal coordinate reference frame is established.

Additional crosslinks within a molecule or to previous molecules may be specified. Crosslinks may even be formed from a given atom to two or more atoms on separate residues. The only information required is the residue numbers and graphical names (unique atom names assigned by the user) of the linking atoms, as well as the molecule number on which the atoms are located. This approach is much better than other methods which have common atoms on two residues that are superimposed when the residues are linked. The latter method assumes advance knowledge the residues to be joined. For an arbitrary collection of residues, this knowledge may not be available.

Again, after the crosslink is made, the additional information for the linking atoms is calculated. This is done only after all molecules have been read in and the new data are stored at the end of the arrays already formed.

The input for the link step of actinomycin is given in Figure 5 and described in Appendix B. The first few cards specify the files on which the six residues are located and their identifying names. The later part of the data divides actinomycin into three molecules; each is linked noncovalently to the third dummy atom at the beginning of the first five residue segment. Covalent linkages are given after the listing of the residue names. For example, atom OG on residue 1-THRP is linked to atom C on residue 5-NMVA of molecule 1.

EDIT

The individual residues now have been linked together in an arbitrary conformation depending on the internal coordinates of the first three atoms of each residue. Before performing calculations on the assembled molecule, it is necessary to assign new coordinate values to at least some of the variables. The main function of EDIT is to perform this task in a convenient manner.

The major source of coordinates for a calculation is from x-ray structure determinations. Therefore, it is important to have a convenient method of transferring the crystallographic coordinates into the program. EDIT will read a structure in Brookhaven protein data bank format and replace the current coordinates with the new values. It accomplishes this by matching atom graphical names on a residue by residue basis. In addition the program can output a file with the same format as the x-ray file and with the original coordinates replaced by the current set or ones from MINM.

The x-ray coordinates for large molecules usually omit hydrogens and sometimes mobile side chains on the surface. EDIT will assign positions f these atoms after the entire x-ray file is read. Bon lengths and angles for the missing atoms are taken from the original internal coordinate values. The dihedral angle of an atom X is found by calculating the dihedral angles of all other atoms bonded to the same atom as X and then placing X in a chemically reasonable position (determined by the valency of the atom to which these atoms are attached).

This process will always work except for the first few atoms of a molecule. Here the user must explicitly read in internal coordinate values for one or two atoms. EDIT warns the user when this is necessary. For proteins, it is possible to use a standard set of cards that will calculate automatically the correct positions of initial hydrogens.

EDIT can also read in new internal coordinates if the coordinates are in internal coordinate format or new Cartesian coordinates if they are in Cartesian format. In the former case, a new internal coordinate can affect the structure of every atom further up the tree. Thus, there is an option to read in internal coordinates for an atom and to convert it immediately to Cartesian coordinates. This will leave the rest of the system unchanged. To facilitate placing the coordinates in the desired format, one-word commands lead from Cartesian to internal coordinate transformations and vice versa. The program keeps track of the coordinate type status of each residue, so it is possible to have a mixture of coordinate types entering EDIT and unnecessary coordinate transformations are prevented.

Other editing functions include the ability to prepare an input file for a CNDO calculation, to replace any or all of the atomic charges, and to change the pucker of any five-membered ring in the system. Moreover, dummy atoms can be omitted.

The final function is the addition of counterions to the molecule. After checking for the absence of a salt bridge or a close contact, the program inserts a counterion a specified distance from a particular atom of a designated residue type. All atoms bonded to the particular atom except E atoms are used to determine where the counterion is placed. These counterions enable any system to be transformed easily from a charged system to a neutral one.

```
1.      TEST FOR INTERCALATOR
2.  20  21
3.   0  0  1  0
4. XYZ
5. OMIT
6. REA
7. XYZBIN 9  /u/pak/acti.XRAY
8. QUIT
```

Figure 6. Sample input for EDIT.

Figure 6 lists the EDIT input for actinomycin. These cards provide a starting guess for the coordinates that have been obtained from the x-ray coordinates. Additional comments are given in Appendix C. The next step is the preparation of actinomycin for minimization.

PARM

PARM assembles various files together. These files include an EDIT output file, a file with the parameter values, and a file with new starting coordinate values. The user may also read in a file with a set of coordinates that will be used as constraints during the minimization. PARM also calculates information that MINM needs, such as which atoms are in the groups for analysis and which atoms are to be constrained. This step is best performed with a separate program so that MINM, which is the only CPU bound step of the process, is as compact as possible.

The primary function of PARM is the connecting of parameters needed for the empirical energy function to the arrays (bond lengths, etc.) that have already been formed. This is accompanied by matching atomic symbols representing bonds, angles, and dihedrals to the types of the atoms comprising the bond, angle, and dihedral arrays. Each atomic symbol is associated with an equilibrium value for the variable and a force constant. Every dihedral term can have as many terms of the Fourier series expansion as desired. When a successful match is made, a pointer is placed into a special array. This avoids recomputing the parameters in MINM over and over.

A generalized angle is represented by X—atom—X and a generalized dihedral by X—atom—atom—X. A successful match only requires the agreement of the middle atoms. A search is then made among the rest of the array for a more specific match of atoms types. In the case of dihedral angles, there are terms consisting of graphical names at the end of the dihedral parameters. These are useful for defining improper torsion angles, e.g., "out-of-plane" dihedral angles used to keep carbonyl oxygens in the amide plane of peptides.

The nonbonded parameters are read in as atom type, polarizability, number of effective electrons, and vdW radius. They are used along with the Slater–Kirkwood formula[2] to determine the A and B constants for a Lennard-Jones 6–12 potential. The A and B parameter values may also be read in, as well as the vdW radii and well depths. The user may select one of several nonbonded parameter sets by reading in a key word that also appears at the beginning of each nonbonded parameter set in the parameter file.

The arrays for the nonbonded parameters and the pointers go up as $n*n/2$ where n is the number of atom types. Since there are a large number of atom types (>40) to handle both nucleotides and proteins, there is an option to declare equivalent atom types for the nonbonded interactions. A redefinition array is used to assign sequential numbers to the unique atom types occurring in each molecule. The parameters of these are then collected into smaller arrays which are passed to MINM. In all molecules run thus far, <20 atom types have occurred at any one time.

The user may read in hydrogen bond 10–12 parameters; the hydrogen bond pair types are taken from this list. The same pointer array is used for both hydrogen bond pairs and regular nonbonded pairs. The two types are distinguished by the use of a negative sign in front of the hydrogen bond pointers. Though the current parameter set does not use 10–12 vdW terms, these pointers are still essential so that regular vdW repulsion terms may be omitted.

Hydrophobic or hydrophilic atom types may be specified in the parameter list. MINM makes use of these types if an option is set to specify a solution potential. This is a potential in which hydrophobic atoms have modified vdW interactions and the charged interactions are greatly reduced.[13]

If no parameter values are found for an atom or variable, an error message is given. A missing parameter value might not be noticeable in the minimization. This is especially true for a molecule with hundreds of atoms, even if all energy terms are printed out.

Special bonds, angles, and dihedrals may be specified as constraint variables. Their ideal values are read along with a force constant. These quantities do not necessarily correspond to atoms bonded together. For example, a special constrained bond might be used to hold a nucleotide base pair together. These additional constraints are particularly useful in reaction path calculations. It is possible to absorb the constraints into the variable arrays in such a way that no special constraint terms are needed in the MINM program.

PARM can read in new coordinates for the minimizer as well as a set of coordinates to be used as constraints. It also prepares the initial nonbonded calculation for MINM by making a list of all nonbonded and hydrogen bond pairs within a specified radius of each atom. MINM is often called to do only one function evaluation to ensure that all input values are correct before proceeding with the minimization. The creation of these lists is much more time consuming than a function evaluation and it is worthwhile minimizing the number of times this is done.

When internal coordinates are to be minimized, PARM prepares the list of variables according to the user's specification. It then automatically identifies dependent dihedral functions. Atoms in loop closings are excluded from becoming dependent functions.

Finally, PARM may be used to specify groups that are involved in energy partitioning in MINM. These groups consist of arbitrary combinations of residues and atoms. A decision was made to place this section here rather than in MINM because the routines use arrays to set up the groups that do not have to be passed to MINM. In addition, PARM is needed to put the new minimized coordinates into the appropriate array before calling MINM.

There is a standard input for this section and MINM that does not need to be changed unless the user wants a special feature. The sample input, which consists of numbers and parameters, will not

```
          PROTEIN TEST
 20          22
  0    0    3.     SCHE
BINBIN

    1
3.2
 0    0
 PARAMETERS          TEST
BR  C    CA  CB  CC  CD  CE  CF  CH  C2  C3  C*  CL  F  H  HC  HO  HS  HW  H2
H3  I    IM  IP  M1  M2  N  NA  NB  NC  NT N2  N3  N*  O  OH  OM  OS  O2  P
S   SH    SF  S*  CU  CT
C    H  HO  H2  H3  IM  IP  N  NA  NB  NC  N2  NT  N2  N3  N*  O  OH  OM  OS
P    O2
C -C* 450.      1.444      THY
C -C3 300.      1.523      END(AA)

X -NT-X      46.5      109.5
C2-CH-CA     46.5      115.      BLEO

X -OS-P -X      3    .75      0.      3.
OH-P -OS-C3     1    .75      0.      2.

C5* -C4* -C3* -01*      7.0      180.      3.

   HO   O   4. 1
   HO   OH   4. 2

C   C*   CB   CC   CA
CD   CE   CF
N      NA   N2   N*   NT

SCHE       SK
  H       0.42       0.9       1.30
  CH       1.35       6.       1.85

END
```

Figure 7. Sample input for PARM.

be given for actinomycin. We do, however, present an abbreviated model of the input to the parameter program (Fig. 7 and Appendix D). At the present time, there are no aliphatic hydrogens in the force field. However, incorporation of these does not involve any program changes. Only the database and the parameter set need be modified. Some preliminary development of a parameter set has been done and it is this set that we will use in the calculations. We can now proceed to the MINM program.

MINM

MINM was designed starting with the empirical energy function program written by Gelin[10] that uses analytical first derivatives. The major differences from the Gelin program are in the choice of minimizer, in the treatment of nonbonded interactions (see below), the possible use of a solution potential, and the ability to have several terms in the Fourier series expansion of the dihedral energy term.

MINM adjusts the coordinates with a modified conjugate gradient minimizer[13] until the energy function is at a relative minimum. The user can assign weights to each term in the energy expression including the 1–4 vdW and electrostatic terms. He also has the option, as mentioned previously, of selecting a solution or gas phase potential.

The most difficult part of an energy calculation is the problem of long-range interactions. In a large molecule, such as a protein, it is not feasible to calculate all nonbonded pairs for each energy function evaluation. The usual solution is to evaluate only the pairs out to a cutoff distance (e.g., 9 Å) from each atom. He can then ask that all interactions between 7.5 and 8.5 Å be multiplied by a cubic equation whose value varies from 1 at the lower bound to 0 at the upper bound. The gap between 9 and 8.5 Å is necessary so that atoms just outside the cutoff will have the opportunity to move inside the region of interaction. Atoms thus appear and disappear smoothly. The interaction list will be reevaluated as often as the user considers necessary. At present this may be done for each function evaluation, only at restarts, or only at the beginning of the calculation.

There is still the problem of dealing with dipoles that have been split by the cutoff radius. Rather than store lists of atoms that make up dipoles, the method of neutral spheres[14] is used. The assumption is made that the interaction shell around each atom is neutral. If it is not, then the charge needed for neutrality is distributed uniformly about the surface of the sphere and one additional electrostatic calculation (that of the atom at the center of the sphere with this charge) is carried out. This method greatly improves the convergence of the electrostatic interactions.[15]

The choice of the dielectric constant is an important decision. The user may select either a constant value or a value multiplied by the distance between the two atoms.[10] The only basis for selecting a dielectric constant is at present empirical. The dielectric constant is, of course, only necessary because the explicit solvent interactions, as well as polarization effects, are neglected.

After computing the energy minimum, MINM may be used for energy partitioning. All inter- and intraenergy components are evaluated for the groups selected in PARM. An evaluation of the results can reveal "hot spots" in the molecule where there is strain or close contact between residues. Salt bridges or regions with many hydrogen bonds will have a large negative electrostatic energy. Along with the partitioning, individual energy terms with values greater than a specified cutoff value are listed. This feature has been useful in identifying mistakes in our choice of dihedral parameters.

ANAL

After minimizing the molecule, it is useful to know what changed and by how much. The analysis of the rms deviation of the calculated and original coordinates is done by the method of Ferro and Hermans.[16] Another subroutine currently available is one that automatically calculates sugar pucker parameters for specified atoms in a particular residue type.

The results are displayed graphically on an E&S Picture System. A color real-time graphics display has been used to compare calculated with initial structures. The visual display with two or more colors dramatically illustrates how the two structures differ, whereas the calculation only tells where they differ.

APPLICATION TO ACTINOMYCIN D

As discussed above, we have used AMBER to examine the conformation and rotational barriers in actinomycin D. To carry out such a calculation, we used the standard amino acid bond, angle, dihedral, and nonbonded parameters from Gelin and Karplus,[10] our set of *ab initio* charges for the amino acids,[17] and CNDO/2 charges for the actinomycin chromophore,[18] with force constants determined by analogy with aromatic rings in the nucleotides and amino acids.* In each case, the energy was completely optimized in Cartesian coordinates (96 atoms, 288 degrees of freedom). A typical calculation took about 1 h on the PDP 11/70. Our convergence criterion was an rms gradient of 0.1 kcal/Å, at which point the energy is converged to better than 0.01 kcal/mol.

The first question we addressed was: What is the relationship between the Sobell model built geometry (built to intercalate into a DNA double helix) and the x-ray structure?[12] We first minimized the Sobell model using a dielectric constant of 1 and then carried out the minimization using a dielectric constant of $4r$. The results were comparable (Table II) so we proceeded with the remaining calculations with the larger dielectric,

* We used a torsional barrier of 8 kcal/mol for rotation around the ester C—O bond and 6 kcal/mol for rotation around the chromophore —C—O bond.

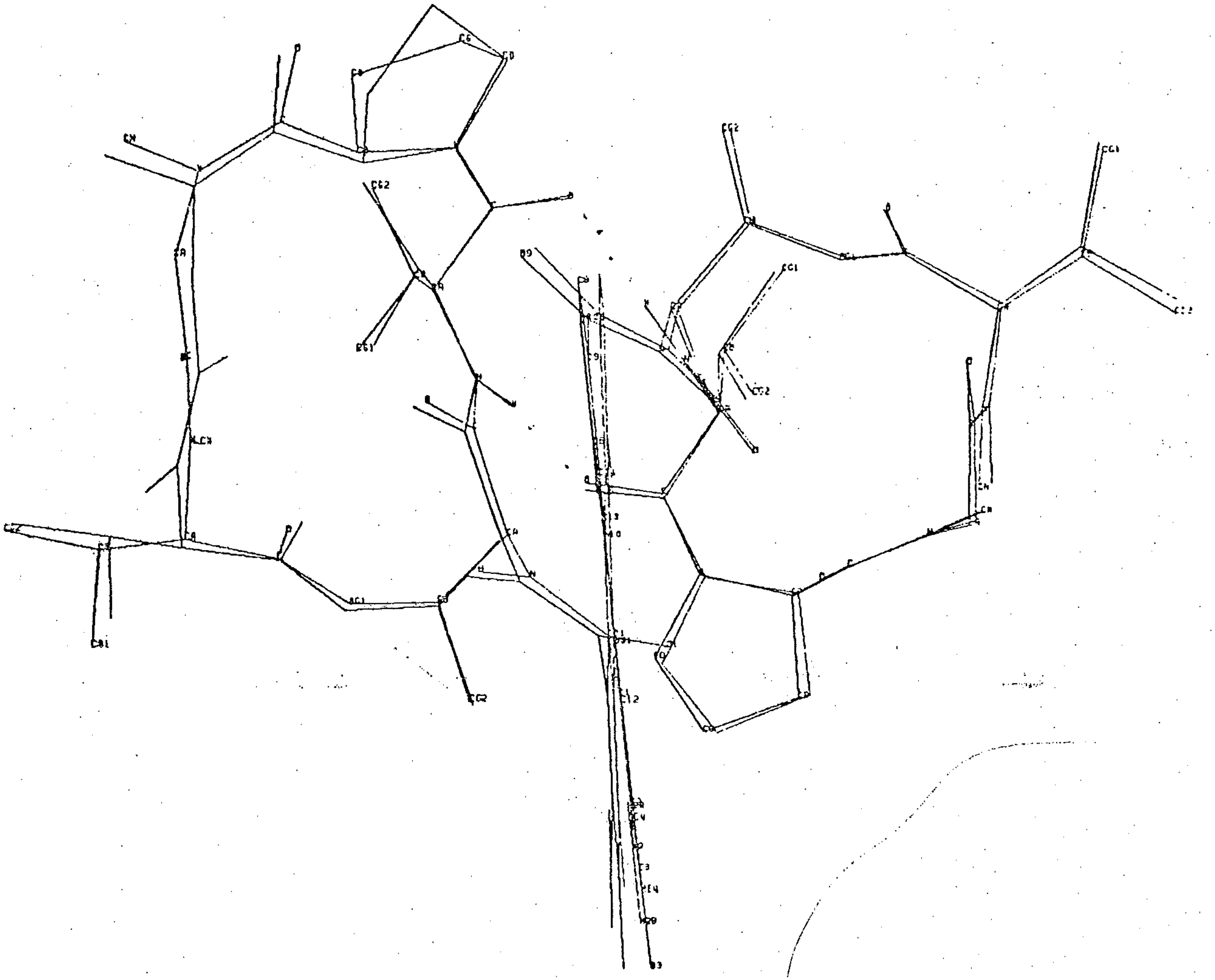


Figure 8. Comparison of minimized x-ray and minimized Sobell model of actinomycin D.

Table II. Energies and structure comparisons for Sobell model and x-ray structures.

	Energies (kcal/mol)							
Structure	Total	Bond	Angle	Dihedral	Elec	vdW	1–4 Elec[a]	1–4 vdW[a]
Sobell model, $\epsilon = 1$, initial	315	61	58	56	−253	−66	316	143
Sobell model, $\epsilon = 1$, final	101	10	22	38	−247	−78	306	50
Sobell model, $\epsilon = 4r$, initial	258	61	58	56	−19	−66	25	143
Sobell model, $\epsilon = 4r$, final	45	9	22	38	−18	−80	24	50
X-ray structure, $\epsilon = 4r$, initial	166	50	31	52	−19	−68	24	96
X-ray structure, $\epsilon = 4r$, final	45	9	21	37	−18	−79	24	51

	rms comparison (Å)				
	1	2	3	4	5
1. Sobell model, initial	—	0.93	0.87	0.62	0.94
2. Sobell model, final, $\epsilon = 1$	0.93	—	0.12	0.99	0.53
3. Sobell model, final, $\epsilon = 4r$	0.87	0.12	—	0.92	0.49
4. X-ray, initial	0.62	0.99	0.92	—	0.70
5. X-ray, final, $\epsilon = 4r$	0.94	0.53	0.49	0.70	–

[a] Atoms that are separated by only three bonds; the program has the option to reduce the weighting of these interactions, but we included them with a weight of 1.0 here.

Table III. Comparison of proline pucker and H bonds.

	Proline pucker[a]			
Structure	q_1^b	ϕ_1^b	q_2	ϕ_2
Sobell model, initial	0.26	65	0.26	65
Sobell model, optimized	0.33	39	0.33	39
X-ray initial	0.39	177	0.31	354
X-ray, optimized	0.37	176	0.34	38

	H bonds			
	r_1^c (H-...O)	θ_1^c (N—H...O)	r_2^c (H...O)	θ_2^c (N—H...O)
Sobell model, initial	1.98	158°	1.98	158
Sobell model, optimized, $\epsilon = 1$	1.77	166	1.77	163
Sobell model, optimized, $\epsilon = 4r$	1.91	162	1.90	164
X-ray initial	1.95	155°	2.01	156
X-ray, optimized	1.97	159°	1.93	159

[a] Using the approach of Cremer and Pople and ordering the atoms N, C_α, C_β, C_γ, C_δ, for example; proline ring of the first peptide for the x-ray structure, energy optimized has a dihedral angle of $C_\alpha C_\beta C_\gamma C_\delta$ of 37° and a dihedral angle $C_\beta C_\gamma C_\alpha N$ of 29°, very close to De Tar and Luthra's "best" structure in their model compound of −35° and 28°.

[b] The proline pucker for the peptide ring connected to the lactone ring with the NH_2.

[c] r_1 and θ_1 refer to the distance and angle from the N—H of the valine of peptide I (off the ring with the NH_2) to the C—O of the valine of peptide II; r_2 and θ_2 are the corresponding quantities from the N—H of D-Val of peptide II to the C—O of the D-Val of peptide I.

since it probably is a more realistic representation of the effective dielectric in solution. We then optimized the x-ray structure and the result of this optimization is also summarized in Table II. Using the graphical features of the program we can easily compare any two structures: the Sobell and x-ray minimized structures are compared in Figure 8 where the major qualitative difference between the two is clearly seen. The two structures differ mainly in the pucker of the proline rings. The Sobell model was built to be twofold symmetric and the proline rings of both peptide rings are in a Pro-II conformation; the x-ray structure has the first proline in a Pro-I conformation, but the second in Pro-II (Table III). According to DeTar and Luthra,[19] Pro-I is about 1 kcal/mol more stable in the model compound they studied. In fact, our refined x-ray structure is about 1 kcal/mol more stable than the Sobell model when it comes to bond angle and dihedral energies (mainly due to this difference in Pro puckers), but the Sobell structure compensates by having about 1 kcal/mol more nonbonded attraction.

The results of the energy component analysis are presented in Table IV for the Sobell optimized geometry and the x-ray optimized geometry.These analyses illustrate the main energetic features of these structures. There is considerable strain in the structures, but also considerable attractive interactions between the peptide rings. The two inter-ring H bonds between the D-Val residues stay in a similar geometry to what they had prior to geometry optimization, even with a dielectric constant of 4r and no explicit H bond function (Table III). In fact, the attractive interaction between the rings is dominated by the nonbonded terms.

A very interesting aspect of the biological activity of actinomycin D is its slow dissocation from double-stranded DNA.[20] Shafer, Burnette, and Mirau[21] have found an activation free energy for dissocation of 21.4 kcal/mol for this process; this energy is sensitive to substitutions on the proline ring (changing to four- and six-membered rings or placing a C═O on C β). We thus decided to examine the hypothesis that the dissociation of actinomycin D from DNA might be coupled to the cis–trans isomerism around the D-Val–L-Pro or L-Pro–Sar amide bonds (both of these bonds are cis in the x-ray or Sobell structures). This is very simple to do with AMBER; one adds a single card to the parameter input, specifying the numbers of the four atoms involved in the dihedral to be constrained, the value of the angle constraint, and the force constant for the constraint. For the rotations around the two amide bonds, we carried out the rotations both clockwise and counterclockwise to a perpendicular (ϕ = 90° or 270°) conformation. We then changed the constraint angle to the initial angle plus 180° to change the cis amide linkage to trans; when this had converged, we removed the constraint so that the molecule would not be forced to have an absolutely planar amide.

The results of such calculations on the Sobell optimized structure for rotation around the two peptide bonds are presented in Table V. It is cl

Table IV. Energy components analysis of the optimized structures (kcal/mol).

Sobell model

Intragroup terms[a] Group	Total	Bond	Angle	Dihedral	Elec	vdW	1–4 Elec	1–4 vdW
Peptide I	31.0	3.7	9.2	16.7	−7.8	−2.05	12.5	17.3
Chromophore	27.4	2.9	3.7	6.1	0.0	−4.7	−1.1	20.5
Peptide II	23.1	2.2	8.9	15.6	−7.7	−20.6	12.2	12.5

Intergroup terms—electrostatic + vdW	Peptide I	Chromophore	Peptide II
Peptide I	—	—	—
Chromophore	−8	—	—
Peptide II	−17	−8	—

X-ray model

Intragroup terms Group	Total	Bond	Angle	Dihedral	Elec	vdW	1–4 Elec	1–4 vdW
Peptide I	30.0	3.7	8.9	15.4	−7.9	−20.6	12.5	18.0
Chromophore	27.5	2.8	3.6	6.6	0	−4.8	−1.1	20.5
Peptide II	23.0	2.2	9.0	15.4	−7.6	−20.6	12.2	12.6

Intergroup terms—electrostatic + vdW	Peptide I	Chromophore	Peptide II
Peptide I	—	—	—
Chromophore	−8	—	—
Peptide II	−16	−8	—

[a] If a particular energy term involves members of more than one group, this energy is divided equally among the groups.

that whereas the barrier for rotation around the Val–Pro bond clockwise is reasonable, and that counterclockwise is too high, the trans arrangement of this peptide is very high in energy, due mainly to strain and to breaking one of the interring hydrogen bonds (with a dielectric of $4r$, the hydrogen bond breaking energy is at most 1 kcal/mol). On the other hand, rotation around the sarcosine bond is far more feasible, with a barrier in the right range and a reasonably low energy for the trans conformation. Thus, if amide bond rotation is essential for DNA dissociation of actinomycin and analogs, it is more likely to be the Pro–Sarc amide bond which is the key. Specific stabilizing forces could stabilize the trans Val-Pro conformation, but the magnitude of the stabilization required (10–20 kcal/mol) makes this unlikely.

There have been two previous studies of the conformation of actinomycin using conformational energy approaches. Ponnuswamy et al.[22] took an initial x-ray structure of Sobell and carried out a two-stage refinement of it, varying dihedral angles after the bond lengths and angles had been standardized. They found an rms difference between the initial x-ray and refined structure of 0.47 Å, which is somewhat smaller than our value of 0.70 Å, but the qualitative features of their structure (trans–cis–cis–trans peptide linkages and interring H bonds) are quite similar to the initial x-ray structure and the optimized x-ray structure presented here.

Table V. Energies for rotation around amide bonds (kcal/mol).

Bond	Direction	ΔE^a	ΔE^b
D-Val–L-Pro	Clockwise	23	20
D-Val–L-Pro	Counterclockwise	29	19
L-Pro–Sar	Clockwise	20	18
L-Pro–Sar	Counterclockwise	21	3

[a] Relative energy (compared to optimized structure) of constraining the peptide bond O—C—N—C_α dihedral angle to be 90° (clockwise) or 270° (counterclockwise).

[b] Relative energy of structure with peptide bond in the trans configuration.

DeSantis et al.[23] carried out model-building studies with some energy refinement and found a low-energy conformation quite close to the x-ray structure, but with an additional H bond between the NH_2 of the lactone ring and the carbonyl of the first pentapeptide ring. He used this H bond to explain the asymmetry of the NMR resonances of the two pentapeptide rings. It is clear, however, from Table V that for the Sobell and x-ray minimized structures (both without this lactone-carbonyl H bond), there are significant differences in the energy and structure of the two pentapeptide chains, which could also rationalize the asymmetry observed in the NMR spectra for the two peptides.

All in all, refs. 22 and 23 were very useful and significant starts on a complete unraveling of the conformation of actinomycin and its interaction with DNA and this effort is a further step in that direction. Our article has the following firsts for actinomycin D: complete energy optimization of the molecule (allowing for proline repuckering and examination of the effect of dielectric constant on the energy minimization) and a study of the process of cis–trans amide isomerism. Important future directions include a comparison of the isomerism process in different analogs and energy calculations on actinomycin–DNA complexes.

SPECIAL FEATURES OF AMBER

Variable Dimensioning

If one fills the arrays with "magic" numbers, there soon comes a time when the user wishes to do the calculation with a molecule that is too large for the program. It is then that the usefulness of variable dimensioning occurs.

The major methods for variable dimensioning are the use of either a large block of storage that is broken into pieces for the arrays or of a macro capability in which variable names are replaced by numbers.

AMBER uses the macro approach, but with a FORTRAN program written by one of the authors (PKW) that makes the substitutions. This program uses a few input values to calculate values for specially named variables in AMBER. The substitutions are made in the FORTRAN program as it is read in line by line and a new FORTRAN program is output that may then be compiled.

By avoiding a system program or the use of special FORTRAN capabilities, this feature is not only made completely general, but special capabilities are present that the other methods cannot give. For example, by setting a flag, arrays that involve internal coordinates are given a value of 1. In the program they are dimensioned to different quantities such as the number of variables. This option is used when minimizing Cartesian coordinates to conserve storage space.

Portability

The program is written entirely in standard FORTRAN with many special features, such as free format input routines, also written in FORTRAN. The UNIX version, running under a FORTRAN 77 compiler, was transformed into an IBM 360 version after a weeks' work by one of us (PKW) with the aid of an interactive editing program. The LBL CDC 7600 Cartesian coordinate version also took about a week to write. Both the IBM and CDC versions use an overlayed LINK program. However, all other programs and the entire UNIX version do not use overlays. The PDP 11/70 system will handle up to 284 atoms; the limits of the other systems have not been reached (700 atoms are easily handled).

Error Checking

The macro capability of AMBER is used to support extensive error checking of array limits. Other useful features are the checks that all variables have parameters and that the tree structures defined in PREP are consistent, e.g., a B atom has at least three atoms attached to it. If there is a special rule that something cannot be done, the program checks that the user does not try to do it. In addition, there are flags that govern how much input and output is printed.

COMPARISON OF PROGRAMS

The reader is justified in asking how this pro-

Table VI. Comparison of empirical energy function programs.

	1	2	3	4	5	6	7	8	9
CAMSEQ-II (ref. 7)	√	no	no	no	√	√	√	no	no
CAMSEQ/M (ref. 8)	√	√	√	√	√	√	√	no	no
Ref. 10	√	√	√	no	√	no	√	no	√
QCFF/PI (ref. 5)	no	no	no	no	no	√	√	√	√
REFINE (ref. 9)	no	√	√	no	√	√	no	no	√
UNICEPP (Scheraga)	no	√	√	no	no	√	no	no	√
AMBER (ref. 15)	√	√	√	√	√	√	√	√	√

1. Modular programs that can fit onto a small computer without overlays.
2. Variable dimensioning of programs.
3. Ability to link residues that have been defined previously.
4. Ability to crosslink atoms in two arbitrary residues merely by giving atom numbers.
5. Ability to read coordinates in protein data bank format and calculate positions of missing atoms.
6. Automatic calculation of variables needed for the energy function.
7. Matching of parameters to variables using atom-type names.
8. Ability to minimize both Cartesian and internal coordinates.
9. Ability to minimize all degrees of freedom.

gram differs from the ones that precede it. A comparison is made in Table VI with other major program packages for which the authors were able to get either an instruction manual or source listing.* An attempt was made to obtain the most current version of each package. However, programs are under continual development and some of the deficiencies marked might now be remedied.

In general the most common deficiencies are the lack of modular design, the inability to crosslink arbitrary residues conveniently, and the inability to minimize both Cartesian and internal coordinates. The last feature is desirable for an efficient study of molecular conformations and interactions.

Programs based on internal coordinates are usually designed to minimize only with respect to dihedral angles. If all degrees of freedom are varied, the process is often carried out in Cartesian coordinates. Often one desires to vary only dihedral angles to explore conformational space thoroughly. At the minimum located, one would like to relax all degrees of freedom without having to go through the time-consuming transformations from internal to Cartesian coordinates.[24] This second step is very important. If only dihedral angles are varied, it has been shown that incorrect conclusions can be reached.[25]

Conformational flexibility of molecules, such as nucleotides, have a dependence on the sugar puckering. One cannot vary only the dihedral angles in the sugar ring and hope to represent the allowed conformations of the nucleotide correctly.[26] It is reasonable to vary only the main dihedral angles if the angle internal coordinates of the sugar are also varied. Therefore, a program that adjusts only internal coordinates should have the capability to vary all degrees of freedom.

CONCLUSION

AMBER is designed as a modular system in which extensive work was carried out to ensure having a database that is sufficiently flexible for use in either Cartesian or internal coordinate minimization, as well as interactive graphics.

It takes approximately a day for a novice user (one who has never done an empirical energy function calculation but knows a little about computers) to become proficient with the major features. The use of standard input files that have already been prepared for PARM and MINM contribute to the short learning time. Another day is required to learn to use PREP. However, improvements underway to simplify the preparation step—automated generation of atom types and of

* An exception was made for CAMSEQ/M. In this case the authors listed the features described in ref. 8.

the basic tree structure—should shorten the learning time.

A completely interactive graphical version has just begun to be written. The applications section is being extended and a link to other quantum mechanical programs is being made.

Current applications of the program include calculations on nucleotide conformations and the use of the program in the crystallographic refinement of bungarotoxin.

Anyone interested in obtaining a copy of the program should contact the authors.

The authors would like to thank the NSF (CHE-7681718) and the NIH (CA-25644) for partial support of this research. This research was also supported in part by the National Resource for Computation in Chemistry under a grant from the National Science Foundation and the U.S. Department of Energy (Contract No. W-7405-ENG-48). Most of the program development was carried out at the UCSF Computer Graphics Lab, Professor R. Langridge, Director, supported by NIH-RR-1081. The authors would also like to acknowledge their scientific debt to the following people: The original FORTRAN program for MINM's Cartesian coordinate energy function with analytical derivatives was developed by the Lifson group at the Weizmann Institute (A. Warshel, Ph.D. thesis, Weizmann Institute, 1969) and first applied to protein conformations by Levitt (M. Levitt, Ph.D. thesis, Cambridge University, 1971). This program was later refined and expanded by Gelin and Karplus (B. Gelin, Ph.D. thesis, Harvard University, 1976). It was one of the authors (PKW) experience at Harvard University (in the Karplus group) with this system of programs that greatly helped in the development of AMBER.

APPENDIX

A. Input for PREP (Fig. 4)

Card 1 is a title card. Card 2 gives the name of the residue and the request to output the final coordinates as internal coordinates. The next card gives the file numbers for the input and output files. Card 4 states that the data is input with the connectivity defined by the tree structure, all dummy atoms (DU) at the beginning of the residue are omitted, and new charges will be read after the coordinate cards.

A loop-closing bond will be assumed between all atoms closer than the distance on card 5 (if the atoms are not already bonded). Card 6 reads in loop-closing bonds explicitly. A blank card ends the input. Card 7 reads in out-of-plane torsion angles. This input also ends with a blank card.

Starting with card 10, the atomic information is input. All input in this section and the following one is in free format. Each card includes the graphical (unique) name and symbolic name (for the parameter selection) of each atom, a symbol describing its position in the tree, the atoms with which the atom forms a bond, bond angle, and dihedral angle (these numbers are not necessary in this input), the values of these quantities, and finally the charges. The input of this section ends with a blank card.

Finally, the charges are read on card 21. As before, the input ends with a blank card.

B. Input For LINK (Fig. 5)

Card 1 is a title card. Cards 2–7 list names of residues that are not in the database and the names of the files with the FORTRAN data produced by PREP. The input ends with a blank card. Card 9 identifies the symbol used for dummy atoms. Card 10 gives flags that identify how much information about the system is output.

Card 11 is a little card for molecule 1. The following card identifies the molecule as an "other" (not protein or nucleotide). The "1" signifies that crosslink information will be input. Card 13 lists the residues of the molecule sequentially. Each residue is identified by a four-letter code. After ending the input with a blank card, card 15 gives the residue number and unique graphical name of the two atoms that are to be crosslinked.

The input for this molecule ends with a blank card. The input for the following two "molecules" is identical with the exception of additional information on cards 18 and 24. The last two fields on these cards give the molecule and atom numbers of the previous molecule to which the first M atom of the present molecule is attached. There is an additional parameter that specifies that the link will be covalent. Since the link between "molecules" does not involve the first M atom of the new "molecule," this option is not used here. The default is a noncovalent link (no additional bond is generated). This is usually done to link all internal coordinates to a common origin. The final card ends all input for LINK.

C. Input for EDIT (Fig. 6)

Card 1 is a title card. The file numbers for the input and output files are given on card 2. Card 3 tells how much of the information to print about the system (coordinates, bond array, excluded atom list, etc.). Card 5 requests a conversion of coordinates to Cartesian format. The next card states that all dummy atoms should be omitted. Card 6 tells EDIT to read new coordinates into the program. Card 7 states that these coordinates will be Cartesian and from a binary file called /u/pak/acti.XRAY. The last card gives the command to quit reading in commands.

D. Input for PARM (Fig. 7)

Card 1 is a title card. Card 2 gives the input and output file numbers. Card 3 identifies the file numbers for the files with new starting coordinates and constraint coordinates. If the file numbers are 0, as they are here, no new files are read. The third field gives the cutoff distance for nonbonded interactions. SCHE is the name of the nonbonded parameter set desired in the calculation. Card 4 states whether the files identified on card 3 are in binary or formatted format. Cards 5 and 6 control the amount of information that the output gives. Card 7 gives the distance below which all hydrogen-bonded pairs are printed. Card 8 identifies whether information about groups is to be read for analysis and whether internal or Cartesian coordinates are to be minimized.

The following group of cards gives the parameter information. Card 9 is a title card for the parameters. Cards 10–12 give the symbols for atomic types. Cards 13 and 14 list the atom types that are hydrophilic (needed for the solution potential). In both cases the input is ended by a blank value of the symbol.

Cards 15 and 16 are the bond length parameters. They list the two atom symbols, the force constant, and equilibrium bond lengths. The input, as in the following sections e, ends with a blank card. Cards 18 and 19 are the bond angle parameters. They include the three atom symbols, force constant, and equilibrium value. An "X" stands for any atom symbol. A match is made with these cards only if a card can not be found with all three atom symbols listed explicitly.

Cards 21–24 list dihedral parameters. The last three numbers give half the barrier height, the phase shift, and the periodicity of the torsional function. If the last value is negative, additional terms in the torsional potential are read. The first number is used to partition the barrier height between the various end atoms involved in rotation around a given bond. Card 24 is also a dihedral parameter, but is listed with graphical names rather than atom symbols. This is done to identify out-of-plane torsion angles.

Cards 26 and 27 give the atomic symbols for atoms that can bond with hydrogen. There is a field for parameters *A* and *B* for a 10–12 vdW potential. These numbers are zero here because the hydrogen bonding is carried out by neglecting the 6–12 vdW term. The next field gives the cutoff for the hydrogen bond interaction. The last field gives the number of the hydrogen bond.

Cards 29–31 gives equivalences in the nonbonded interactions. Cards 33–36 are the nonbonded input. Card 33 lists the name of the nonbonded set and a symbol (SK) that shows that Slater–Kirkwood parameters are to be read. The following cards give the atom symbol, atomic polarizability, effective number of electrons, and the vdW radius. Card 37 ends the input.

References

1. N. Allinger, *J. Am. Chem. Soc.*, **99**, 8127 (1977); E. M. Engler, J. Andros, and P. V. R. Schleyer, *J. Am. Chem. Soc.*, **95**, 8005 (1973); S. Fitzwater and L. Bartell, *J. Am. Chem. Soc.*, **98**, 5107 (1976); S. Lifson and A. Warshel, *J. Chem. Phys.*, **49**, 5116 (1968).
2. A. T. Hagler, E. Huler, and S. Lifson, *J. Am. Chem. Soc.*, **96**, 5319 (1974); F. A. Mormany, L. M. Carruthers, R. F. McGuire, and H. A. Scheraga, *J. Phys. Chem.*, **78**, 1594 (1974); M. Levitt, Ph.D. thesis, Cambridge University, 1971.
3. W. K. Olson and P. J. Flory, *Biopolymers*, **11**, 1, 25, 57 (1972); O. Matsuoka, C. Tsai, and E. Clementi, *Biopolymers*, **17**, 33, 51, 67 (1978); V. Sasisekharan and A. V. Lakshminarayanan, *Biopolymers*, **8**, 505 (1969).
4. N. L. Allinger, J. T. Sprague, and T. Liljefors, *J. Am. Chem. Soc.*, **96**, 5100 (1974).
5. A. Warshel and M. Levitt, *Quantum Chemistry Program Exchange, No. 247*, Chemistry Department, Indiana University, Bloomington, IN.
6. M. Browman, A. Burgess, L. Dunfield, S. Rumsey, and H. Scheraga, *Quantum Chemistry Program Exchange, No. 361*, Chemistry Department, Indiana University, Bloomington, IN.
7. R. Potenzone, Jr., E. Cavicchi, H. J. R. Weintraub, and A. J. Hopfinger, *Comput. Chem.*, **1**, 187 (1977); H. J. R. Weintraub, and A. J. Hopfinger, *Int. J. Quantum Chem. Quantum Biol. Symp.*, **2**, 203 (1975).
8. H. J. Weintraub, in *Computer-Assisted Drug Design*, E. C. Olson and R. E. Christoffersen, Eds., *Am. Chem. Soc. Symp. Ser. 112*, Am. Chem. Soc., Washington, D.C., 1979, p. 353.
9. J. Hermans, D. R. Ferro, J. E. McQueen, and S. C. Wei, *Proc. Symp. Environmental Effects on Structure and Properties, Jerusalem, 1975*, Reidel, Dordrecht, 1976, p. 459.
10. B. R. Gelin, Ph.D. thesis, Harvard University, Cambridge, MA, 1976; B. R. Gelin and M. Karplus, *Biochemistry*, **18**, 1256 (1979).
11. J. Hermans, Jr. and D. Ferro, *Biopolymers*, **10**, 1121 (1971).
12. S. C. Jain and H. M. Sobell, *J. Mol. Biol.*, **68**, 1, 21 (1972).
13. P. Weiner and M. Karplus, unpublished work.
14. D. J. Adams, *Chem. Phys. Lett.*, **62**, 329 (1979).
15. P. Weiner and P. Kollman, CECAM Workshop on Simulation of Enzyme Catalysis, Paris, Summer 1979.
16. D. R. Ferro and J. Herman, *Acta Crystallogr. Sect. A*, **33**, 345 (1977).
17. D. M. Hayes and P. Kollman, *J. Am. Chem. Soc.*, **98**, 3335 (1976).
18. M. E. Nuss and P. Kollman, *J. Med. Chem.*, **22**, 1517 (1979).
19. D. F. Detar and N. Luthra, *J. Am. Chem. Soc.*, **99**, 1232 (1977).
20. W. Muller and D. Crothers, *J. Mol. Biol.*, **35**, 251 (1968).
21. R. Shafer, R. Burnette, and P. Mirau, *Nucl. Acids Res.*, **8**, 1121 (1980).
22. P. Ponnuswamy, R. McGuire, and H. A. Scheraga, *Int. J. Prot. Pep. Res.*, **5**, 73 (1973).
23. P. DeSantis, R. Rizzo, and G. Ughetto, *Biopolymers*, **11**, 279 (1972); *Nature New Biol.*, **237**, 94 (1972).
24. C. Pottle, M. S. Pottle, R. W. Tuttle, R. J. Kinch, and H. A. Scheraga, *J. Comput. Chem.*, **1**, 46 (1980).
25. R. E. Christoffersen, in *Computer-Assisted Drug Design*, E. C. Olson and R. E. Christoffersen, Eds., *Am. Chem. Soc. Symp. Ser. 112*, Am. Chem. Soc., Washington, D.C., 1979, p. 3; **101**, 6842 (1979).
26. M. Levitt and A. Warshel, *J. Am. Chem. Soc.*, **100**, 2607 (1978).