Distribution
Category UC-405

SAND93-8688
Unlimited Release
Printed November 1993

A Genetic Algorithm Based Method for Docking Flexible Molecules

Center for Computational Engineering - 8117
Sandia National Laboratory
Livermore CA 94551-0969
email: rsjuds@ca.sandia.gov

E.P. Jaeger and A.M. Treasurywala
Sterling-Winthrop Inc.
1250 South Collegeville Road
Collegeville, Pennsylvania 19426-0900

Abstract

We describe a computational method for docking flexible molecules into protein binding sites. The method uses a genetic algorithm (GA) to search the combined conformation/orientation space of the molecule to find low energy conformations. Several techniques are described that increase the efficiency of the basic search method. These include the use of several interacting GA sub-populations or niches; the use of a "growing" algorithm that initially docks only a small part of the molecule; and the use of gradient minimization during the search. To illustrate the method, we dock Cbz-GlyP-Leu-Leu (ZGLL) into thermolysin. This system was chosen because a well refined crystal structure is available and because another docking method had previously been tested on this system. Our method is able to find conformations that lie physically close to and in some cases lower in energy than the crystal conformation in reasonable periods of time on readily available hardware.

I. Introduction

One of the most sought after goals of computer aided drug design is the ability to design a ligand that strongly binds to a biologically important receptor. This is a complex problem but one that can be broken down into several smaller ones, some of which have been solved but most of which are still open. The major pieces are experimentally determined binding site geometries, methods to dock ligands into proteins, and accurate force fields to quantitatively predict binding energies. High resolution protein structures that define binding sites are becoming available using either x-ray crystallographic or solution NMR techniques. Several methods are being developed to search the ligand-protein conformation space to find energetically favorable binding geometries for specific ligands. Force fields that describe the internal motion of the proteins and the interaction of ligands with proteins have been developed, but they still suffer moderate to serious problems of accuracy. Models of the important water layer about the protein-ligand complex have been and are being developed, but much work still needs to be done to increase accuracy and computational efficiency. In addition to these requirements, a long term need is the ability for the computer to design ligands by itself, i.e. to search molecule space as well as conformation space.

This paper presents a new approach to the ligand-protein docking problem. A number of methods have been reported in the literature[1-18], each of which treats some aspects of the problem. A complete model of ligand-protein docking requires; (1) ligand flexibility; (2) protein flexibility; (3) variable positioning of the ligand; and (4) full protein-water-ligand interactions. The DOCK method of Kuntz, et al.[8,9,11,16,17] for instance treats a simplified model in which ligand and protein flexibility are ignored. Our present method treats the following model. The ligand can be freely positioned and is fully flexible. Full protein-ligand interactions are used. However, the protein is held rigid and only crystallographically determined waters are included. We give a more complete comparison of our method to others in the literature in the Conclusion section. We use a genetic algorithm[19-24] (GA) to guide the conformational search. The GA method has been used by a number of groups for conformation searching of model systems[25], small molecules[26,27], proteins[28-30] and DNA[31], and it has been largely successful. In particular, the method appears to be faster than both simulated annealing[25] and directed search[27]. Other chemical

applications of the GA that have been reported include protein[32] [33]and polymer[34] folding, 2D NMR peak assignments[35], alloy modeling[36] and pharmacophore elucidation[37].

The method draws on our GA-based small molecule conformation work[27], with a few modifications to make docking practical. First, we define a pivot atom in the ligand to serve as the origin for translating and rotating the molecule. Second, to increase computational efficiency, we screen all conformations with a fast "bump count" potential that rejects all conformations in which the ligand penetrates into the protein. Third, we use a "growing" algorithm in which only a small portion of the ligand is initially docked. This sub-molecule contains the pivot atom and its nearest neighbors. As the search proceeds, atoms are added to the ligand until it has grown to its full extent.

We begin by laying out the basic method for applying GAs to the ligand docking problem. We then present the results of a series of computations where we dock Cbz-GlyP-Leu-Leu (ZGLL) into thermolysin. This system is chosen because a well refined crystal structure is available[38-40] and because another docking method has previously been tested on this system[7]. Our method is able to find conformations that lie physically and energetically close to the crystal conformation. The major purpose of the runs is to learn how to adjust GA search parameters to increase the efficiency of the search. A series of recommendations is presented as well as some future modifications that we plan to incorporate into the method.

II. Computational Approach

Our approach to docking is similar to our basic conformational search method described previously[27]. A GA is used to generate a large number of conformations that are ranked based on energy. During the course of the search, conformations are found with increasingly lower energy. In this section, we briefly describe the workings of the GA and our search approach.

A. Genetic Algorithm

Genetic Algorithms[19-24,41,42] provide a method for finding optima in high dimensional search spaces. In the next sub-section, we describe the particular conformation space we will search over, while here we concentrate on the basics of the GA itself. The GA evolves a population of strings which represent conformations. Strings compete to enter a breeding pool based on their fitness, which corresponds to the energy of the conformation they represent. Over a period of many generations, successively more fit

individuals (i.e. conformations with lower energy) evolve through the process of selection, breeding and mutation. The GA provides a mechanism for efficiently but coarsely searching conformation space. The result at the end of the evolution process is a set of conformations that can be analyzed and in particular, can be gradient minimized.

The form of the GA we use is a modified, binary encoded version of Mühlenbein's breeding GA method[41,42]. When searching for an optimal molecular conformation, a GA population of size N will consist of N sets of M variables which we describe here as a set of torsional angles, $[\theta_1,...,\theta_M]$. Each set of M angles prescribes a molecular conformation whose energy can be evaluated. Each angle is stored in gray coded binary representation and the binary string representing the set of M angles is referred to as a chromosome. Each of the N chromosomes is M times W bits long, where W is the word size, or number of bits used to encode a single angle. Increasing the number of bits increases the resolution of the search. The number of values an angle can assume is 2^W, so increasing W increases the number of values to be considered for each of the M angles and reduces the difference between values differing in only 1 bit by a factor of 2.

A chromosome's fitness is evaluated in a series of steps. First each chromosome is decoded into a set of torsional angles. The molecule to be optimized is then built with the variable torsion angles set to values defined by the chromosome. The energy of the conformation is calculated and is assigned to the fitness, where the lower the energy, the higher the fitness. Once the fitness of each chromosome has been evaluated, the operations of selection, reproduction, crossover breeding and mutation are carried out in order to create a new population for the next generation. We use either step function selection in which all parents in the top $P\%$ of the population, based on fitness, have equal probabilities of being selected to enter the breeding pool; or roulette wheel selection in which parents are chosen for the breeding pool with a probability based on their fitness. A parent who has rank i out of N individuals has a probability of $N(E_{max} - E_i)/\left[NE_{max} - \sum_{i=1}^{N} E_i\right]$ of being selected, where E_i is the energy of parent i and E_{max} is the maximum, or worst energy currently in the population. The reproduction operator exactly reproduces a copy of a chromosome in the current generation so that it will appear in the next generation. Crossover breeding trades subsets of angles between two chromosomes. For instance if a crossover point were chosen between the bits for the ith and $(i+1)$st angles, the two parent chromosomes:

$$[\alpha_1,...,\alpha_i,\alpha_{i+1},...,\alpha_M] ,$$

$$[\beta_1,\ldots,\beta_i,\beta_{i+1},\ldots,\beta_M]$$

would produce the two child chromosomes with angle values:

$$[\alpha_1,\ldots,\alpha_i,\beta_{i+1},\ldots,\beta_M] ,$$

$$[\beta_1,\ldots,\beta_i,\alpha_{i+1},\ldots,\alpha_M] .$$

Mutation, if it occurs, causes a bit to be changed from its current value (0 or 1) to the alternate value (1 or 0). The mutation rate is defined as the probability that a given bit will flip. For instance, if a mutation rate of 0.1 is used with a 10 bit chromosome, each chromosome will have on average one mutation each generation.

The initial population is generated by using a random number generator to set each bit in each chromosome to either 0 or 1. Populations can be comprised of sub-populations known as niches, which have different initial chromosomes. Each niche proceeds independently which allows for checking dependence on initial conditions. Within a niche, a bit is converged when a preset fraction of the chromosomes has the same value in a given bit position. For example, with a convergence threshold of 0.8 and a population of size 10, if there was a bit position that had the same value (0 or 1) in at least 8 of the chromosomes then that position would be converged. Full convergence is reached when all bits are converged.

A feature that we find to be important is niche interaction. In the case that several niches are running in parallel, it helps all of them if they occasionally pass information back and forth. The way this is done is that at specified generations, before the new sub-populations are formed, the best individual in each niche is passed to each of the other niches. These replace the worst individuals in each niche. This is helpful because an individual sub-population will tend to converge onto one or two regions. By introducing a few individuals with good fitness which can lie in other regions of conformation space, the search space can broaden, with the result that new and often better regions can be found.

To summarize, a generation consists of the following steps. The fitness of each chromosome is evaluated. The chromosomes are then ranked in order of fitness from best to worst. In the elitist mode (always used here) the best chromosome from one generation reproduces a duplicate child for the next generation. If niche interaction is to occur in the current generation, the best individual in each niche is passed to each other niche. Next, parents are selected for crossover breeding to fill the remainder of the next generation.

Pairs of parents are randomly selected from the available pool for crossover breeding to produce the next generation's chromosomes. The mutation operator then acts on these chromosomes. The fitness of each individual is once again calculated and the process repeats until the requested number of generations have transpired or the population has converged. There are a variety of stopping criteria that one can use some of which are discussed in Ref.26.

B. Conformational Search Strategy

Here we describe the translation from the generic GA chromosome to a molecular conformation and from there to an energy or fitness. We treat the case of a flexible molecule or ligand docking into the binding pocket of a rigid protein. A conformation is defined by the position and orientation of the ligand as a whole and the values of its dihedral angles. Bond distances and angles are typically fixed during the search phase.

We define one atom in the ligand as the "pivot" and one in the protein as the "target". These should be chosen from some knowledge that the pivot and target atoms will lie close to one another in the actual bound conformation. A reference position is chosen for the pivot atom, lying close to its expected position in the bound conformation. A conformation is built as follows, starting from a constant reference conformation. Each of the dihedrals is set to the value prescribed by the chromosome. The direction of the dihedral is specified so that atoms on the pivot side of the bond being rotated do not move. Once the dihedrals are positioned, the ligand as a whole is rotated about the pivot atom using 3 Euler angle values. Finally, the ligand is positioned in the binding site by placing the pivot atom at the reference position and then adding a specified offset $(\delta x, \delta y, \delta z)$. The Euler angles span the range from 0-360. In our numerical calculations, we let the offset variables span the range $-1.5 < \delta x < 1.5$, etc. This ensures that all conformations of the ligand are "interesting" in that they lie close to the binding site. The chromosome is a bit string that codes for the 3 offset values, the 3 Euler angles and the specified number of dihedral angles.

Once the conformation is defined, the energy can be calculated. For all conformations, we first calculate a "bump energy". This is defined as

$$E_{bump} = 1000 N_{bad} - 10 N_{good} + 100000 - 10(R_{PT} - R_0)^2 \qquad (1)$$

where N_{bad} is the number of "bad" Van Der Waals contacts (defined below), N_{good} is the number of "good" contacts, R_{PT} is the distance from the pivot to the target, and R_0 is the reference distance. To calculate the number of good and bad contacts, each ligand atom - protein atom distance is calculated. If that distance is closer than an inner cut off, N_{bad} is incremented. If the distance is between the inner and outer cutoffs, then N_{good} is incremented. The inner cutoff is defined to be 0.4 times the sum of the Van der Waals radii of the two atoms and the outer cut off is defined as 1.25 times the same sum. The performance of the method is relatively insensitive to the precise values of the numerical parameters in Eq. (1). The main requirements are that conformations that penetrate into the protein are heavily penalized and that the best values of the bump energy are greater than the values of the MM energy calculated for non-interpenetrating conformations. The rational behind calculating the bump energy is that it is fast and provides a good diagnostic of whether a conformation will have a high MM energy because of bad non-bonded contacts.

If a conformation has no bad contacts, then the full MM energy is calculated and returned as the fitness. We use the CHARMm[43] force field as implemented in CCEMD, a C-language molecular dynamics/molecular mechanics program which is based on the MD code of Windemuth, et al.[44] To speed up the computations, no bond, angle or dihedral energy terms are calculated for fixed atoms and no fixed-atom/fixed-atom non-bonded terms are calculated. If the MM energy is sufficiently low, we also have the option to perform a gradient minimization of the conformation and to return the minimized energy as the fitness.

A further modification to the basic method which is vital to its efficiency is what we call "growing". During the early part of the search, few conformations will be found that can slip into a typically narrow binding site, and much effort will be wasted. The idea behind growing is that a small part of the ligand around the pivot atom is docked initially. This sub-molecule is small enough that the GA will quickly find a few good conformations. After a period of time, the ligand is grown by adding the nearest neighbors of the atoms currently included, and the now slightly larger sub-molecule is allowed to search for low energy conformations starting from, but not limited to, a set of reasonable conformations. The growing procedure continues until the entire ligand is included, but the GA search continues on for many generations after the growing procedure terminates. The growing algorithm is implemented as follows. The initial sub-molecule is made up of the pivot atom and its nearest neighbors. All other ligand atoms are added to the non-bonded

exclusion list so that even if they penetrate into the protein, no energy penalty is incurred. This is true for both the bump energy and the MM energy. However, the full chromosome is used and all dihedrals are manipulated. This sub-molecule is used for the first "grow period" generations. At the end of each grow period, the nearest neighbors of each of the currently included atoms are deleted from the non-bond exclusion list and another set of "grow period" generations is run.

Pseudo code for the logic of the main loop of the docking program is given in Figure 1. The only piece that has not yet been described is the analysis stage. Once the grow phase of the search is complete, we save to disk every conformation that passes the bump count test. The set of orientation and dihedral values and the conformational energy are also saved. At the end of the search phase, the energy file is read and conformations are sorted by energy. Duplicates are discarded, where two conformations are considered to be duplicates if their energies are within 1 kcal/mol and their conformational distance is within 5°. The conformational distance between conformation *a* and *b* is defined as

$$r_{conf}^{ab} = \frac{1}{N}\sqrt{\sum_{i=1}^{N}(\theta_i^a - \theta_i^b)^2} \qquad (2)$$

where for convenience the offset distance and the Euler angles are included in the sum. Next, each unique conformation within 40 kcal/mol of the best found during the GA search is gradient minimized (this number is designated Nbset) and a pdb file containing just the minimized ligand conformation is written to disk with its final energy. These files are then further analyzed using Sybyl[45].

III. Test Problem Definition

For our numerical tests, we dock the molecule ZGLL (Cbz-GlyP-Leu-Leu) into thermolysin. The input files for our computations were prepared as follows. The crystal structure of ZGLL in thermolysin was taken from the Brookhaven Protein Data Bank (pdb5tmn.ent). All modifications to the structure were performed in QUANTA.[46] Each residue with at least one atom within 12 Å of any atom in the crystal conformation of ZGLL was retained and all others were deleted. This includes crystallographically determined waters, i.e. nearby waters are included and others are deleted. The remaining residues were capped with COOH or NH_2 groups. All polar hydrogens were added. GLU 143 and HIS 231 were protonated as described in Ref.7. All aryl hydrogens were added. This was

found to be necessary to correctly define the fine details of the shape of the binding pocket and the ZGLL molecule and to reproduce proper binding geometries. Default charges from QUANTA were used. At this stage, pdb and psf files were produced and all subsequent calculations were performed using CCEMD. We used infinite cutoffs and a distance-dependent dielectric with a coefficient of $2r$.

The next step was to relax the orientations of the waters whose hydrogens had been added by QUANTA in arbitrary directions. All atoms in the protein and ZGLL were fixed and the waters were allowed to reorient, using conjugate gradient minimization. After this, the protein and waters were fixed and ZGLL was relaxed to get a reference energy. The ZGLL ligand moved only very slightly from its crystal conformation. The energy of the relaxed crystal structure was -75.1 kcal/mol. In all of our calculations, the protein and associated waters are treated as fixed, and only the ZGLL ligand is allowed to move. In Figure 2, we show the structure of ZGLL and its rotatable dihedrals. Each of the non-rotatable dihedrals was fixed in the trans. orientation.

IV. Numerical Results

The phosphorus atom in ZGLL is defined as the pivot atom and the Zn atom in thermolysin is defined as the target atom. See Section IIB for a description of the pivot and target atoms. The nominal pivot-target distance is 3.25 Å and the nominal position of the pivot is (51.73,18.97,-6.10) which is the crystal position of the phosphorus atom. The pivot atom was allowed to move ±1.5 Å from the nominal position in each direction (x,y,z). The Euler angles and the dihedrals were allowed to rotate over the entire range 0-360°. We used 10 bits to represent each variable, so the resolution is 0.003 Å in the pivot positioning and is 0.35° in each of the angle variables . The search space has 20 degrees of freedom - 6 for the overall position and orientation and 14 dihedral angles. Therefore the chromosomes each contain 200 bits. Several runs were performed to determine how the search variables affected the efficiency of the search. These variables include: (1) the population size; (2) number of niches; (3) grow period; (4) use of gradient minimization during the search; (5) convergence criteria for the gradient minimization; (6) selection method; and (7) the selection and mutation rates. Table 1 summarizes the input parameters for the runs and some measures relevant to the performance. Each run was allowed to proceed for 500 generations except for one case noted in the table. Niche interaction occurred every 50 generations.

From Table 1, we can already draw some conclusions about what helps and what does not in the search, based on the final lowest energy found in each run. The relevant energy is Ebest (2) from Table 1, i.e. the energy after final gradient minimization. We discuss other criteria below. For instance, going from 1 niche to 4 always yields lower final energies which can be seen by comparing the run pairs (4,1), (8,9) and (12,13). The two runs in each pair differ only in that the first uses 1 niche and the second uses 4. In each case there is a large decrease in the best energy found and an increase in the number of low energy conformations found when 4 niches are used instead of 1. To test if the effect is just one of having 4 times as many individuals, we compared runs 2 and 4 which only differ by population (100 vs. 400). Both used a single niche. Here the results are interesting, but difficult to explain because the larger population actually did significantly worse. This shows that the search is sensitive to initial conditions so that starting with multiple populations is important. Both runs effectively converged relatively quickly to one or two regions of conformation space; the larger run just happened to not find one of the lower energy regions.

By far the worst run is number 3 where no growing was done. We tried several other combinations of variables without growing and none of them helped. So far we have been unable to get the method started without growing. We tried longer grow periods, up to 10 generations, but they did not work significantly better than a period of 4. Table 1 also shows that some degree of gradient minimization of "useful" candidate structures during the search phase is helpful. Otherwise, the energies that are passed back as the fitness function can be dominated by one or a few exceptionally high energy interactions. These could be easily alleviated with a few steps of gradient minimization. However it is not essential in our experience to pass back the coordinates of the minimized structure or to minimize to completion. Performing a few steps of minimization proved in our hands to be the best compromise between a time consuming complete minimization and a potentially misleading energy resulting from no minimization at all. However, minimization during the search phase can be very expensive, increasing the search cost by a factor of between 5 and 30.

The step function selection method works better than roulette wheel selection. This is due to the fact that in a population with fitnesses of widely varying magnitudes, the lowest energy conformations will dominate. (For instance energies of conformations vary between -80 and +10000.) These will quickly take over the population, and drive it into a local minimum. Roulette wheel selection is best used on fitness landscapes with less

variability in magnitude than we have here. Low selection rates are better than high rates. This seems to be due to the fact that in a given generation, only a small fraction of the individuals are viable, meaning that they pass the bump count test. This fraction is between 10 and 20% typically. High selection rates have the effect of diluting the contribution to the gene pool of the best 10-20% of the parents by including more low fitness, non-viable parents in the breeding pool. These parents are less likely to produce viable children and they therefore slow down the rate of evolution. The effect of each of the search variables are summarized in Table 2. From this we arrive at the conclusion that it is best to : (1) use niching; (2) use growing; (3) use a few steps of gradient minimization during the search phase; (4) use step function selection; and (5) use relatively low selection and mutation rates.

In any attempt to dock a guest molecule into a host, one is faced with questions about how to best evaluate the results. In the preceding discussing we evaluated an individual docked structure based solely on the final energy. However, there are other criteria for assessment in this case because we know the "right" answer, from the experimental crystal structure. Obviously, a successful run would be one where the docking algorithm converged, in a reasonable amount of time, to the single experimental crystal conformation. Several factors separate reality from this ideal scenario. In most real cases, even where an x-ray structure of the protein-ligand complex exists, it is not clear if the experimental solution is unique or instead if several energetically degenerate states exist from which the crystallization or soaking conditions have simply selected one. Thus a non-unique result from the run need not be viewed as a failure of the method. One important question that can be asked is whether a run finds the particular docked mode that had been observed in the crystal structure. It is safe to assume that this configuration of protein and ligand will represent a low energy state even though it may not be the only one possible.

We now expand our analysis of the runs to include 4 criteria, including the final best energy which has already been discussed. These are: (1) the energy of the best docked conformation (as the protein/ligand complex) from the run. This was always compared to the energy of the crystal structure (=-75.13 kcal/mol). (2) The minimum all atom RMS deviation of the ligand from its known crystal structure. This involved all of the atoms in the ligand that were used during docking (including hydrogens) in our test system (41 atoms). This criterion was used to see how close the run came to actually visiting the crystal conformation at any point during the run, regardless of energy. (3) The minimum torsional RMS deviation of the ligand from the known crystal structure. This was a

measure of the same criterion as in point 2 above but in torsional space instead of Cartesian space. (4) The overall efficiency of the run as measured by the total CPU time used. We have observed that the analysis of the combination of the two RMS criteria is often better than either one alone. Sometimes when torsions are sequentially arranged in a linear array of rotatable bonds, a slight deviation of a value for a rotatable torsion in a bond that is early in the sequence will result in great apparent movement of all atoms past that point in the molecule. This will be reflected in a large all atom RMS even though the two conformations are clearly very similar to the eye. Correspondingly, large compensatory deviations can occur in adjacent torsions to produce conformations that have low all atom RMS values but relatively large torsional RMS values. It is only by viewing both these measure that we find that we get a good overall measure of the "similarity" of two conformations to one another.

In Table 3, we rank each of the runs against the 4 criteria and also give 2 composite criteria. The score on which the first composite is ranked is the sum of the ranks of the 4 criteria, energy, all atom RMS, torsional RMS and total CPU time used. The score on which the second composite is ranked is the same except that CPU time is ignored. For instance, the Composite 1 score of run 1 is 2+3+3+9=17 and the composite score of run 2 is 6+9+8+8=31. Therefore run 1 ranks above run 2. There are 4 runs that rank in the top 4 using both composite scores, those being runs 1, 6, 9 and 11. All use the grow algorithm; all but run 6 use 4 niches, all but run 1 use minimization during the search phase, and all use step function selection. Runs 11 and 6 use the high selection/mutation rate. With the exception of run 6, the composite rankings substantially validate the conclusions summarized in Table 2 of the effect of search variables on the success of the run. Run 6 is anomalous in several ways, and its success may be due to chance. It found very few reasonable conformations, but some of those happened to be low in energy and have small RMS deviations.

Several interesting facts emerge from this analysis. None of the runs actually converged on the crystal structure. This is not unexpected (see above). However, one run (9) found solutions that had better overall energies than the known crystal structure as assessed in the same force field. Thus it may be concluded that the docking method is working. However, it may still be necessary to further calibrate the force field or to choose another one. This issue is currently being addressed. In Figure 3, we show all of the conformations found during run 1 which were within 10 kcal/mol of the best conformation found (which was 2 kcal/mol higher in energy than the crystal conformation). Plots for the other runs look essentially the same. The basic structure is conserved but there are a wide

variety of conformations of the phenyl end which can move out away from the protein. In the force field used here, this group moves into vacuum; presumably with a solvent model, this hydrophobic group will tend to move out of solution and back towards the surface of the protein. Likewise, the carboxyl terminus of the ligand displays quite a bit of flexibility. Guida, et al.[7] found similar flexibility in their lowest energy conformations (c.f. Figure 2 of Ref.7).

Figures 4a-4d are scatter plots showing the correlation between final energy and either all atom RMS deviation or torsional RMS deviation from the crystal structure. Plots are given for the 4 best runs 1, 9, 6 and 11. From these plots we see that there is at most a very weak correlation with all atom RMS and practically no correlation with torsional RMS. There are no points in the torsional RMS plots with values near zero. In all conformations found, the orientation of the peptide O-C-N-H group at the phenyl end was reversed, so that the amide hydrogen pointed into the protein and the carbonyl oxygen pointed away, rather than the reverse seen in the crystal structure. This results from several torsional angles being shifted significantly from their values in the crystal conformation. The last two torsion angles before the phenyl ring are also very variable. In Table 4, we compare the energy and RMS criteria for the best structures in each category for runs 1 and 9. The fact that there are 2 or 3 different conformations in both cases underscores the lack of correlation between RMS deviation and energy. In fact the best all atom RMS conformations are 15 and 23 kcal/mol higher in energy than the lowest energy conformations in the two runs. In Figures 5a-5e, we display the conformations tabulated in Table 4, superimposed on the crystal conformation. In all cases, the computed backbone is close to the crystal backbone, but the phenyl and carboxyl ends vary substantially. One can also see the reversed conformation of the peptide group at the phenyl end.

V. Conclusions

In this paper we have demonstrated the feasibility of using a genetic algorithm search method to dock flexible ligands into protein binding sites. For the test case considered, our method was able to find conformations lower in energy than the crystal conformation. The major purposes of this paper were to describe the method, demonstrate feasibility and explore how the method's efficiency was affected by changing selected search parameters. Several other issues need to be addressed before the final usefulness of the method can be measured.

The first is the accuracy of the potentials, which will in large part determine whether computed low energy conformations (which we find) correspond to actual low energy conformations. This is part of a larger problem of defining good potentials for proteins and ligands, but we see one aspect as being more important than others and this is the treatment of water. Here the only waters we treat are those determined crystallographically. Guida, et al.[7] found that if those water were removed, it was necessary to at least include a continuum water model. We are currently building such a model into our code to test this observation. It may still be necessary to explicitly include a few important bound waters, provided that it is known that their positions are independent of the ligand being docked. Guida, et al. also found that protein flexibility was required, which is a feature we do not include. To increase the selectivity of the method, we may also need to include more accurate charges on the protein.

A second issue is one of speed. As reported here, our method can find conformations close to the crystal structure, although not the crystal conformation itself, in about 40 CPU hours on a workstation (see Run 1). This is acceptable for studying one or a few compounds but is problematic if thousands of compounds are to be screened as in the work of Kuntz and coworkers[8,9,11,17]. The two approaches we are investigating are simply using a larger computer by porting the code to a massively parallel machine; and improving the basic efficiency of the serial version. We have not optimized the number of generations run, and in fact, most of the runs did not make significant progress after about 300 generations. We also have not looked at the effect of changing the number of bits in the binary chromosome which defines the resolution of the search. The resolution we use here is probably unnecessarily high. More efficient energy functions could be implemented, such as the grid methods used by Kuntz, et al.[8,9,11,17]

A considerable amount of work is being done in the docking area and it is interesting to compare the functionality of the methods reported in the literature. In Table 5, we list each of the methods of which we are aware along with whether they include ligand flexibility, protein flexibility, and full orientational motion of the ligand. From the table we see that there are essentially four groups: (1) those that dock rigid ligands into rigid proteins; (2) those that dock flexible ligands into rigid proteins; (3) those that dock flexible ligands into flexible proteins; and (4) those that perform conformational search on ligands in a protein pocket but do not vary overall translation and rotation of the ligand. The present method falls into category (2) but can be extended to include protein flexibility by adding torsional motion of selected protein sidechains to the GA chromosome. However, it

is an open question whether this added expense is justified in the absence of more realistic protein-ligand potential functions.

Acknowledgments: RSJ acknowledges partial support by the Department of Energy under contract DE-AC04-76DP00789 and by the Structural Biology Initiative at Lawrence Livermore National Laboratory. AMT and EPJ wish to acknowledge the support of Dr. John Wendoloski for helpful and supportive discussions during the development of this work.

Run	Pop	Niche	Grow	Min.	Select/ mutate	Search CPU	Analyze CPU	Nbest	Ebest (1)	Ebest (2)	Best AARMS	Best TRMS
1	100	4	4	n	0.2/0.02	2058	396	453	-67.8	-73.5	1.26	53.9
2	100	1	4	n	0.2/0.02	605	288	351	-49.0	-53.1	2.20	77.9
3	100	1	0	n	0.2/0.02	466	187	44	348.7	63.0	3.06	76.0
4	400	1	4	n	0.2/0.02	2333	532	296	1.4	-24.9	2.54	73.1
5	100	1	4	n	0.5/0.05	95	7	2	47.6	-11.2	2.23	96.7
6	100	1	4	y [a]	0.5/0.05	323	22	11	32.9	-56.9	1.14	65.1
7	100	1	4	y	0.5/0.05	567	10	5	53.7	-22.7	1.56	91.1
8	100	1	4	y	0.2/0.02	21369	411	592	-55.6	-55.8	2.04	72.0
9	100	4	4	y	0.2/0.02	17270	641	1205	-70.7	-80.0	1.05	50.4
10	100	4	4	n	0.5/0.05	384	5	2	105.7	-38.5	1.67	99.4
11	100	4	4	y	0.5/0.05	2579	101	71	-43.4	-62.5	1.41	33.1
12	100	1	4	n	roul/0.02	91	3	1	417.2	25.1	2.16	107.5
13	100	4	4	n	roul/0.02	377	11	5	45.9	-26.8	2.62	107.8

Table 1 - Parameters for the 13 runs performed. Pop is the population. Niche is the number of niches each of size Pop. Grow is the grow period (see text). Min. says whether or not gradient minimization was performed during the search phase. (a): In run 6, a smaller number of minimization steps (5) was used than in the other runs (10). Select/Mutate are the selection and mutation rates. roul indicates that roulette wheel selection rather than boxcar selection was used. Search and Analyze CPU are the times (in minutes) used for the two phases on an SGI R4000 Indigo. Nbest is the number of unique unminimized conformations found within 40 kcal of the minimum energy conformation before gradient minimization. Ebest (1) gives the lowest energy found before final gradient minimization and Ebest (2) the lowest energy after gradient minimization. The two conformations are typically not the same. The energy of the relaxed crystal conformation is -75.13. Best AARMS gives the best all atom rms distance from the relaxed crystal structure. Best TRMS gives the best torsion angle rms from the relaxed crystal structure. Neither of these are necessarily the same conformation as Best E (2). All runs ran for 500 generations except for 9 which was stopped after 130.

GA variable	Run Pairs	Summary of effect
Population size (100 vs. 400)	(2,4)	Ambiguous
Number of niches (1 vs. 4)	(4,1) (8,9) (12,13)	Adding niches always helps
Grow period (0 vs. 4)	(3,1)	Growing is required
Use of gradient minimization (no vs. yes)	(1,9) (5,7) (2,8)	Gradient minimization always helps but is very expensive
Gradient minimization conv. (loose vs. tight)	(6,7)	Ambiguous
Selection method (step function vs. roulette wheel)	(2,12) (1,13)	Step function always better than roulette wheel
Selection and mutation rates (0.2/0.02 vs. 0.5/0.05)	(1,10) (2,5) (8,7) (9,11)	Lower rates always help

Table 2 - A summary of the effect of GA run variables on the minimum energy found.

Run	Energy	AARMS	TRMS	CPU	Composite 1	Composite 2
1	2	3	3	9	3	2(t)
2	6	9	8	8	7	5
3	13	13	7	7	13	10(t)
4	9	11	6	11	10 (t)	7
5	11	10	10	2	8 (t)	8
6	4	2	4	3	1	3
7	10	5	9	6	6 (t)	6(t)
8	5	7	5	13	6 (t)	4
9	1	1	2	12	2	1
10	7	6	11	5	5	6(t)
11	3	4	1	10	4	2(t)
12	12	8	12	1	8 (t)	9
13	8	12	13	4	10 (t)	10(t)

Table 3 - Ranking of the runs by criteria other than the best energy found. The first 3 columns give the ranks of the runs based on the best conformation found, as measured by energy, all atom RMS deviation from the crystal conformation and torsional RMS deviation from the crystal. The fourth column gives the rank based on total CPU time (search + analysis) used, where the run using the least amount of time is rank 1. Composite 1 is the rank based on the sum of columns 1 through 4; Composite 2 is the rank based just on the sum of columns 1 through 3, i.e. with CPU time neglected. (t) indicates a tie.

	Conformation	Energy	AARMS	TRMS
run 1	Best Energy	-73.5	1.44	65.9
	Best AARMS	-58.2	1.26	62.3
	Best TRMS	-70.8	1.52	53.9
run 9	Best Energy	-80.0	1.90	81.8
	Best AARMS	-57.3	1.05	50.4
	Best TRMS	-57.3	1.05	50.4

Table 4 - A comparison of the conformations that had the best rank for the 3 criteria of final energy, all atom RMS (AARMS) and torsional RMS (TRMS) for runs 1 and 9.

Method	Trans/Rot	Ligand Flexibility		Protein Flexibility		References
		Torsions	All Terms	Torsions	All Terms	
GA search	yes	yes	yes	no	no	Present Work
Brownian Dynamics	yes	no	no	no	no	2,14
Systematic Search	no	yes	no	no	no	12
Monte Carlo (Macromodel)	no	yes	no	no	no	7
Annealed Dynamics	yes (?)	yes(?)	yes(?)	yes(?)	yes(?)	6
Steric Fitting (DOCK)	yes	no	no	no	no	8,9,11,17
Molecular Dynamics	yes(?)	yes	yes	yes	yes	3,18
Misc. Hybrid Methods	yes	yes	yes	no	no	4
	no	yes	yes(?)	no	no	10,13,15
	yes	yes	yes	yes	yes	5
Distance Geometry	yes	yes	yes(?)	no	no	1

Table 5 - A comparison of docking methods reported in the literature. This list is not meant to exhaustive. It is merely intended to illustrate the different strategies that have been used to dock small molecules into macromolecules. A question mark indicates that the method could have the ability indicated, but that the reference either did not use the ability or did not say it was used.

References:

(1) Blaney, J. M., Dixon, J.S. *Ann.Repts.in Med.Chem.* **26**, 281-285 (1991). and references cited therein.

(2) Allison, S. A., Northrup, S.H., McCammon, J.A. *J.Chem.Phys.* **83**, 2894 (1985).

(3) Banci, L., Schroder, S., Kollman, P.A. *Proteins, Str.&Func.* **13**, 288 (1992).

(4) Freeman, C. A., Catlow, C.R.E., Thomas, J.M., Brode, S. *Chem.Phys.Lett.* **186**, 137-142 (1991).

(5) Gallion, S., Ringe, D. *Protein Engng.* **5**, 291-300 (1992).

(6) Goodsell, D. S., Olson, A.J. *Proteins* **8**, 195-202 (1990).

(7) Guida, W. C., Bohacek, R.S., Erion, M.D. *J.Comp.Chem.* **13**, 214-228 (1992).

(8) Kuntz, I. D., Blaney, J.M., Oatley, S.J., Langridge, R., Ferrin, T.E. *J.Mol.Bio.* **161**, 269-288 (1982).

(9) Kuntz, I. D. *Science* **257**, 1078-1082 (1992).

(10) Leach, A. R., et al. *J.Comput.Chem.* **13**, 730-748 (1992).

(11) Meng, E. C., Schoichet, B.K., Kuntz, I.D. *J.Comp.Chem.* **13**, 505-524 (1992).

(12) Meyer, D., Naylor, C.B., Motoc, I., Marshall, G.R. *J.Comput.-Aided Mol.Des.* **1**, 3 (1987).

(13) Moon, J. B., Howe, W.J. *Tetrahedron Comp.Method.* **3**, 697-711 (1990).

(14) Northrup, S. A. *Proc.Natl.Acad.Sci. (USA)* **89**, 3338 (1992).

(15) Rotstein, S. H., Murcko, M.A. *J.Comput.-Aided Mol.Des.* **7**, 23 (1993).

(16) Shoichet, B. K., Kuntz, I.D. *J.Mol.Bio.* **221**, 327-346 (1991).

(17) Shoichet, B. K., Stroud, R.M., Santi, D.V., Kuntz, I.D., Perry, K.M. *Science* **259**, 1445-1450 (1993).

(18) Stoddard, B. L., Koshland, D.E. *Proc.Natl.Acad.Sci. (USA)* **90**, 1146 (1993).

(19) *Proc. of Third Intl.Conf. on Genetic Algorithms*; Schaffer, J. D., Ed.; (Morgan Kaufman, San Mateo, Calif., 1989).

(20) *Proc. of Fourth Intl.Conf. on Genetic Algorithms*; Belew, R. K., Booker, L.B., Ed.; (Morgan Kaufman, San Mateo, Calif., 1991).

(21) Davis, L. *Handbook of Genetic Algorithms*(Van Nostrand Reinhold, New York, 1991).

(22) Goldberg, D. *Genetic Algorithms in Search, Optimization, and Learning*(Addison Wesley, Reading, Mass., 1989).

(23) Holland, J. H. *Scientific American* **July**, 66-72 (1992).

(24) Lucasius, C. B., Kateman, G. *Chemometrics and Intelligent Lab. Systems* **19**, 1-33 (1993).

(25) Judson, R. S., Colvin, M.E., Meza, J.C., Huffer, A., Gutierrez, D. *Intl.J.Quant.Chem.* **44**, 277-290 (1992).

(26) McGarrah, D. B., Judson, R.S. *J.Comp.Chem.* (1993). (in press).

(27) Judson, R. S., Jaeger, E.P., Treasurywala, A.M., Peterson, M.L. *J.Comp.Chem.* (1993). (in press).

(28) Legrand, S., Merz, K. *J.Global.Opt.* **3**, 49-66 (1993).

(29) Dandekar, T., Argos, P. *Protein Engineering* **5**, 637-645 (1992).

(30) Tuffery, P., Etchebest, C., Hazout, S. Lavery, R. *J.Biomol. Str. and Dyn.* **8**, 1267-1289 (1991).

(31) Blommers, M. J. J., Lucasius, C.B., Kateman, G., Kaptein, R. *Biopolymers* **32**, 45-52 (1992).

(32) Unger, R., Moult, J. *J.Mol.Bio.* **231**, 75-81 (1993).

(33) Sun, S. *Protein Sci.* **2**, 762-785 (1993).

(34) Judson, R. S. *J.Phys.Chem.* **96**, 10102-10104 (1992).

(35) Wehrens, R., Lucasius, C., Buydens, L., Kateman, G. *J.Chem.Inf.Comput.Sci.* **33**, 245-251 (1993).

(36) Smith, R. W. *Comp.Phys.Comm.* (1992). (in press).

(37) Payne, A. W. R., Glen, R.C. *J.Mol.Graphics* **11**, 74-91 (1993).

(38) Abola, E. E., Bernstein, F.C., Bryant, S.H., Koetzle, T.F., Weng, J. *Data Commission of the Int'l Union of Crystallography*(Bonn/Cambridge/Chester, 1987).

(39) Holden, H. M., Tronrud, D.E., Monzingo, A.F., Weaver, L.H., Matthews, B.W. *Biochemistry* **26**, 8542 (1987).

(40) Bernstein, F. C., Koetzle, T.F., Williams, G.J.B., Meyer, E.F., Brice, M.D., Rodgers, J.R., Kennard, O., Shimanouchi, T., Tasumi, M. *J.Mol.Bio.* **112**, 535 (1977).

(41) Mühlenbein, H. In *Foundations of Genetic Algorithms*; G. J. E. Rawlins, Ed.; (Morgan Kaufman, San Mateo, Calif., 1991).

(42) Mühlenbein, H., Schomisch, M., Born, J. *Parall.Comp.* **17**, 619 (1991).

(43) Brooks, B. R., Bruccoleri, Olafson, B.D., States, D.J., Swaminathan, S. Karplus, M. *J.Comp.Chem.* **4**, 187-217 (1983).

(44) Windemuth, A., Schulten, K. *Mol.Sim.* **5**, 353-361 (1991).

(45) Sybyl, 6.0 Tripos Associates (St. Louis, MO, 1993);

(46) QUANTA/CHARMM, Molecular Simulations, Inc. (Waltham MA, 1993);The results published were generated in part using the program QUANTA. This program was developed by Molecular Simulations, Inc.

Figure Captions:

Figure 1 - Logic for the GA search.

Figure 2 - Structure of ZGLL with rotatable dihedrals defined.

Figure 3 - A plot showing all conformations found during run 1 that had energies within 10 kcal/mol of the lowest energy structure after final gradient minimization. The crystal conformation is represented by heavy lines. Note the large degree of flexibility at the phenyl end and at the carboxyl group.

Figure 4 - Scatter plots showing RMS deviation from the crystal structure vs. final energy for all structures which were gradient minimized. The left panel uses torsional RMS and the right panel uses all atom RMS. (a) Run 1; (b) Run 9; (c) Run 6; (d) Run 11. Note that there are no points with torsional RMS close to zero. See text for a discussion of this point.

Figure 5 - Plots showing a number of individual conformations superimposed on the crystal conformation (dark lines). (a) Lowest energy conformation from run 1; (b) conformation with best all atom RMS, run 1; (c) conformation with best torsional RMS, run 1; (d) conformation with lowest energy, run 9. This conformation also had the lowest energy for any run in the set. (e) Conformation with lowest all atom and torsional RMS, run 9.

```
generate initial population
setup exclusion list for growing
for(i=1;i=Number of generations) {
     if(i=multiple of grow period) grow ligand one shell
     for(j=1,j=number of niches) {
          evaluate conformations {
               for(k=1;k=Population size) {
                    generate conformation k from chromosome k
                    fitness[k] = bump energy
                    if(N_bad=0) {
                         fitness[k] = MM energy
                         if(MM energy < threshold and minimization on)
                              fitness[k]=gradient minimized MM energy
                         save conformation and energy to disk
                    }
               }
          }
          sort population by fitness
          if(i=niche interaction generation)
               exchange best individuals between niches
          reproduce best individual
          perform selection to define breeding pool
          perform crossover breeding to form balance of new population
          perform mutation on new population except best individual
     }
}
analyze results {
     sort all saved conformation by energy
     delete duplicates
     gradient minimize all conformations with energy < set value
}
```

Figure 1 - Logic for the GA search.

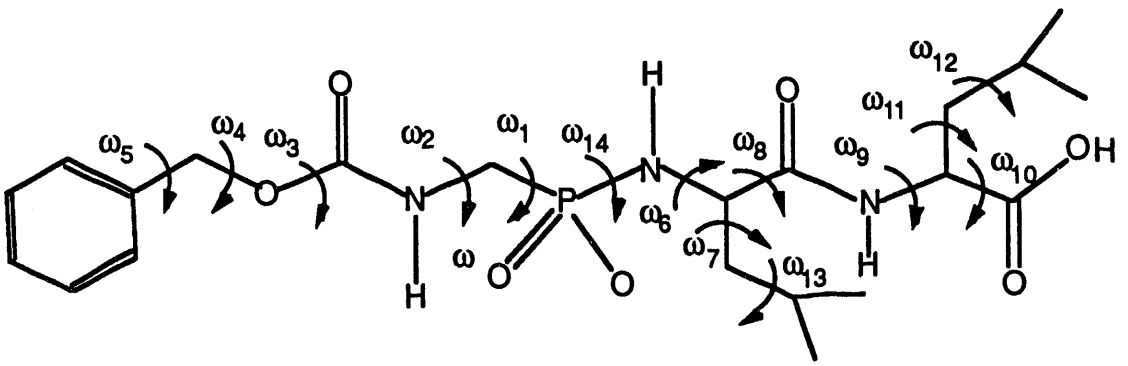


Figure 2 - Structure of ZGLL with rotatable dihedrals defined.

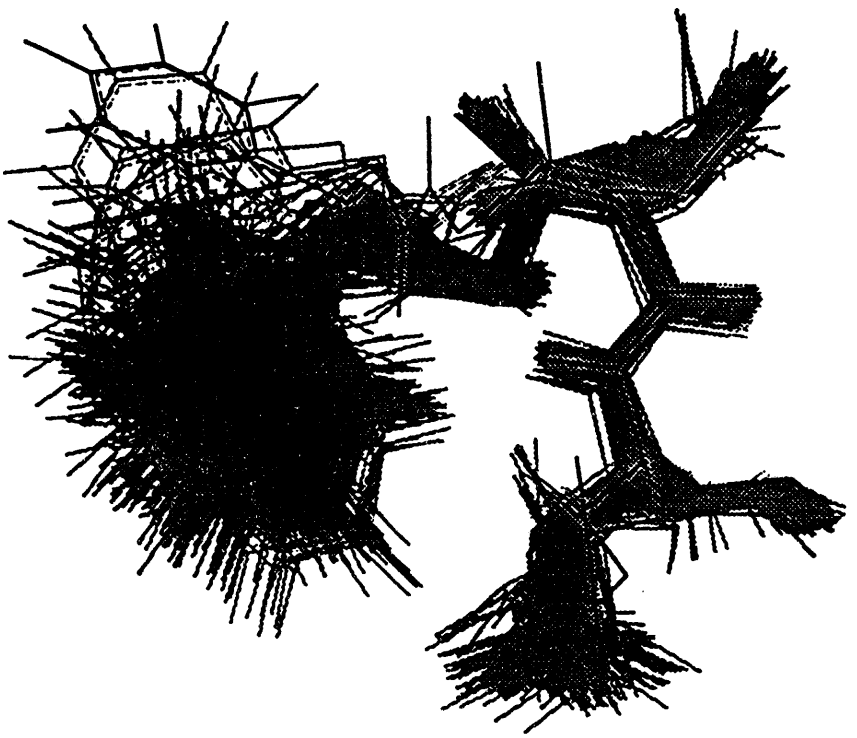


Figure 3

Figure 4a

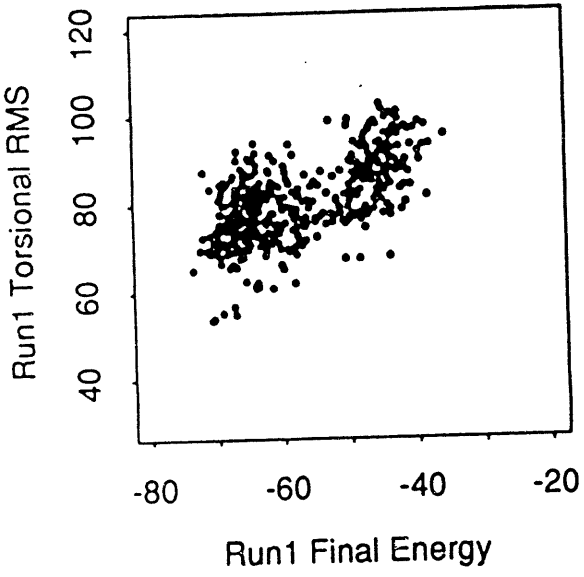
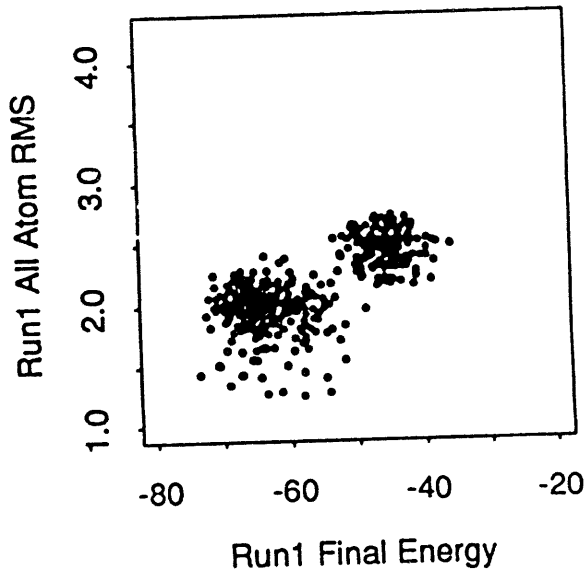
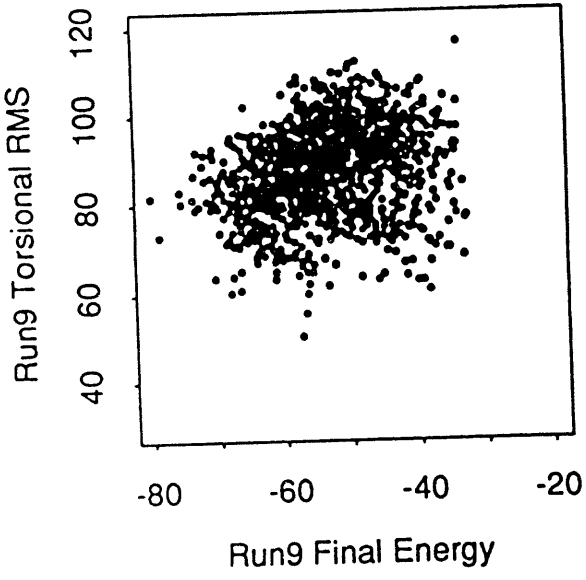


Figure 4b





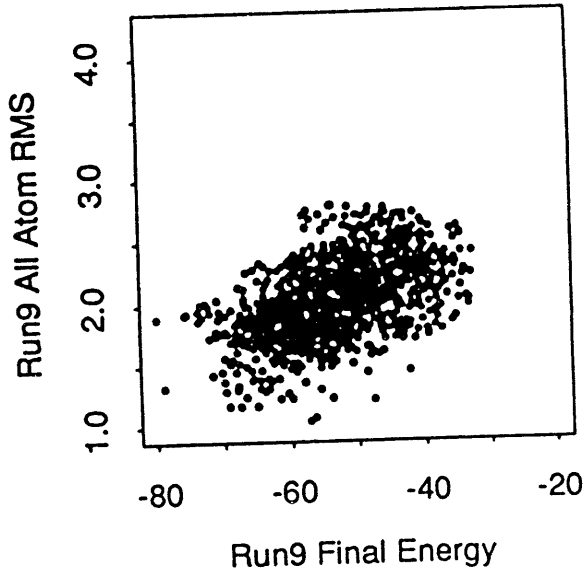


Figure 4c

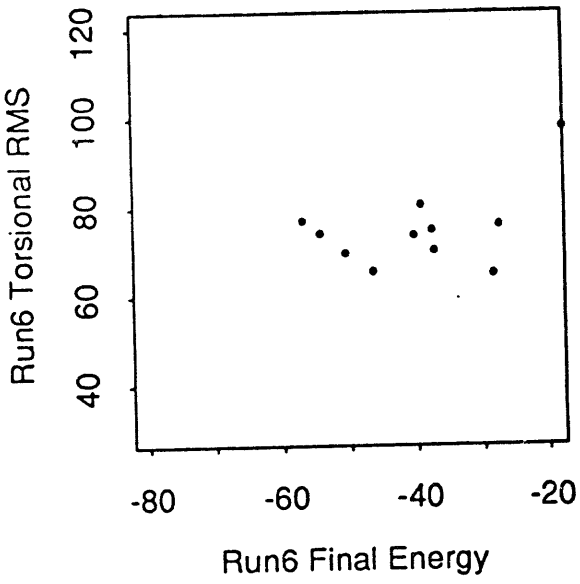
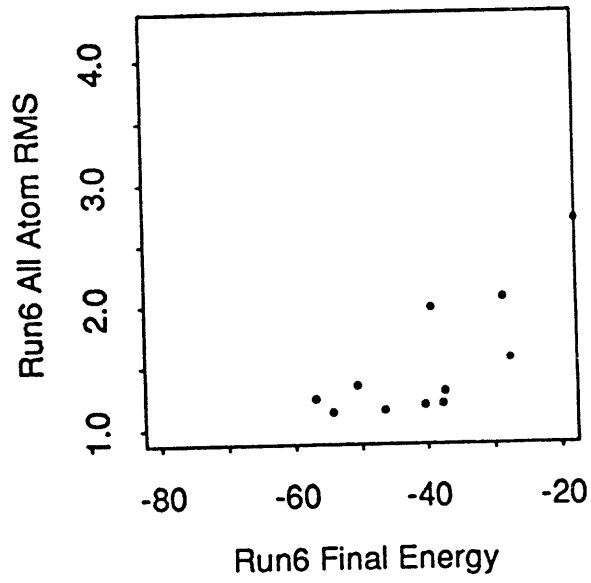
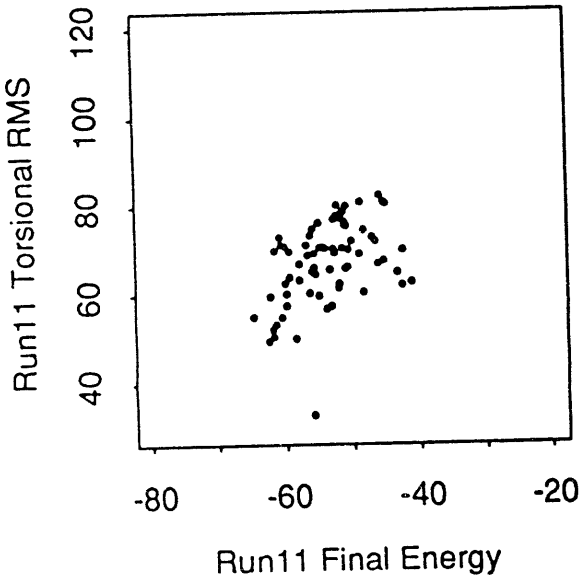
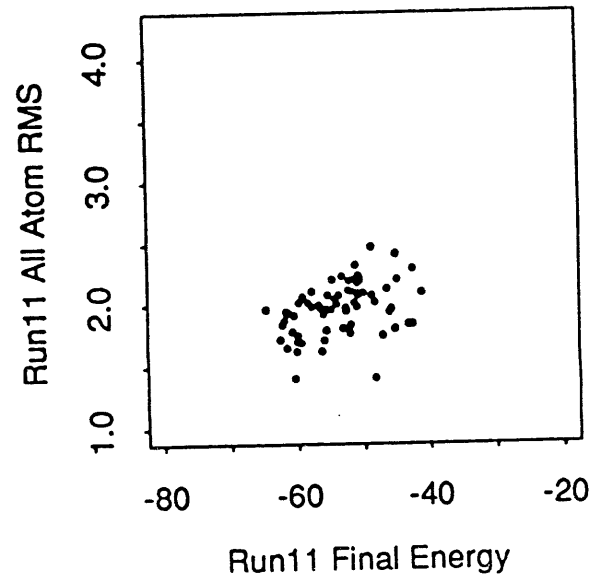


Figure 4d







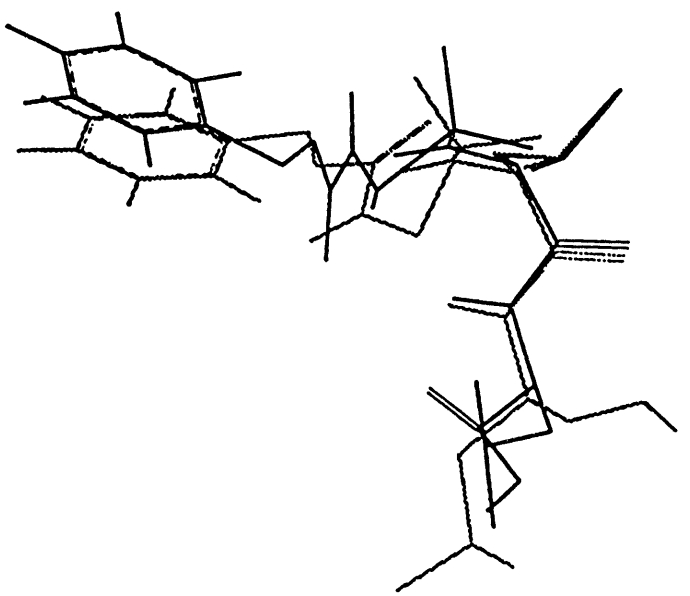


Figure 5a

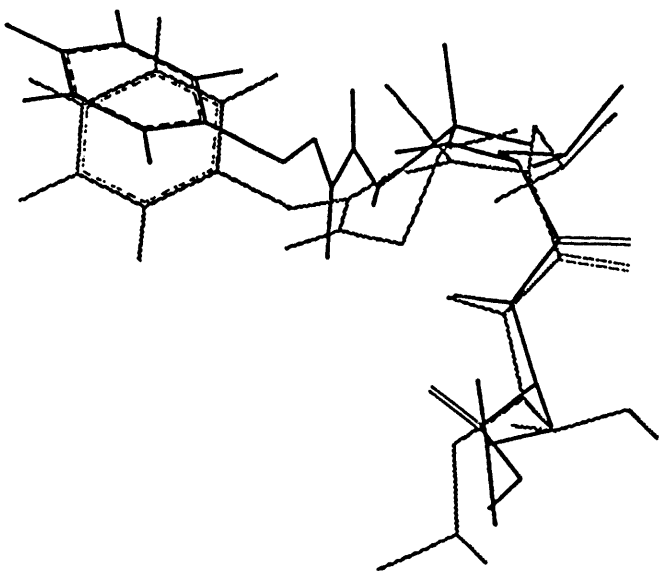


Figure 5b

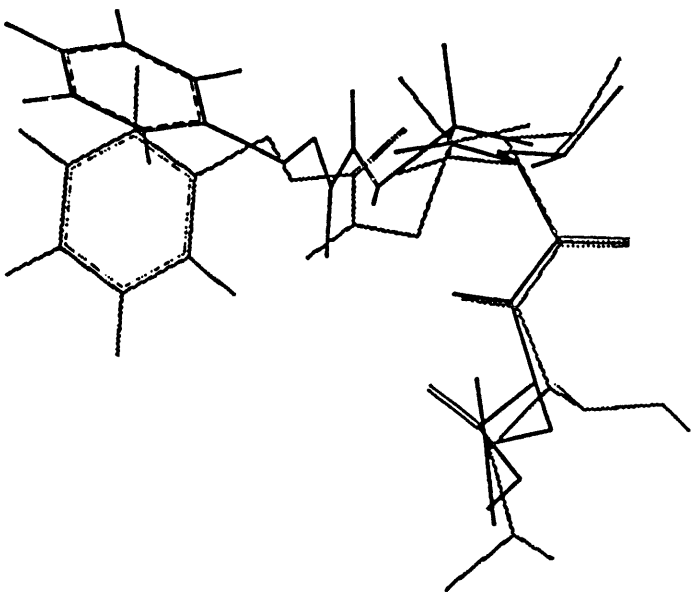


Figure 5c

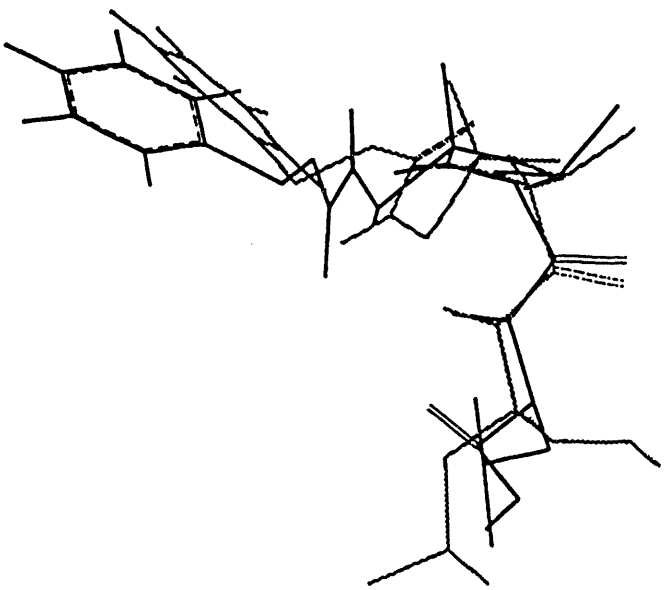


Figure 5d

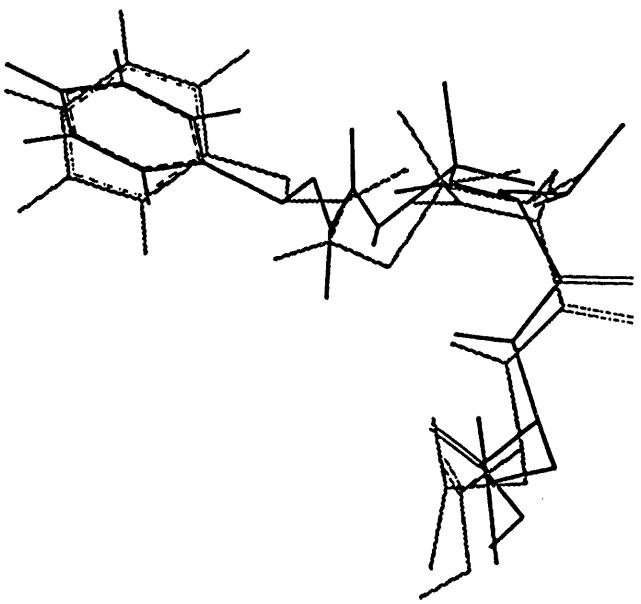


Figure 5e

UNLIMITED RELEASE
INITIAL DISTRIBUTION

8000	J. C. Crawford		
	Attn:	5200	G. E. Ives
		5300	J. B. Wright
		8200	R. J. Detry
		8300	W. J. McLean
		8400	L. A. Hiles
		8500	P. E. Brewer
		8600	L. A. West
		8700	R. C. Wayne
		1900	D.L. Crawford
8100	M. E. John		
	Attn:	8101	T. M. Dyer
		8102	J. Vitko
		8102	M. Lapp
		8103	S. C. Johnston
		8104	R. L. Rinne
		8111	G. Thomas
		8112	R.M. Wheeler
		8113	J.C. Swearengen
		8114	R.J. Gallagher
		8115	M.H. Rogers
		8116	R. Bierbaum
8117	C.L. Bisson		
8117	M. E. Colvin		
8117	C. L. Janssen		
8117	R. S. Judson (50)		
8117	W.P. Kegelmeyer		
8117	C. F. Melius		
8117	J. C. Meza		
8117	L.M. Napolitano		
8117	C. H. Tong		
8354	M. Koszykowski		
8535	Publication for OSTI (2)		
8535	Publications/Technical Library Processing, 3141		
3141	Technical Library Processes (3)		
8524-2	Central Technical Files (3)		

END

DATE
FILMED
4/12/94