A Comparison of a Direct Search Method and a Genetic Algorithm for Conformational Searching

J. C. Meza[1], R.S. Judson[2]
Scientific Computing Department, MS 9214
Sandia National Laboratories
Livermore, CA 94551-0969

T.R. Faulkner
Scalable Systems Division
Intel Corporation

A.M. Treasurywala
Allelix Biopharmaceuticals
6850 Goreway Dr.
Mississaugoa, Ontario L4V 1P1

ABSTRACT

We present results from the application of two conformational searching methods: genetic algorithms (GA) and direct search methods for finding low energy conformations of organic molecules. GAs are in a class of biologically motivated optimization methods that evolve a population of individuals where individuals who are more "fit" have a higher probability of surviving into subsequent generations. The parallel direct search method (PDS) is a type of pattern search method that uses an adaptive grid to search for minima. Both methods found energies equal to or lower than the energy of the relaxed crystal structure in all cases, at a relatively small cost in CPU time. We suggest that either method would be a good candidate to find 3-D conformations in a large scale screening application.

Keywords: global optimization, molecular conformation, nonlinear programming.

[1]email: meza@ca.sandia.gov
[2]email: rsjuds@ca.sandia.gov

1. Introduction

An important goal of computational chemistry research is the design of molecules for specific applications. Factors that have to be taken into account include shape, size, electronic properties, and reactivity. For many physical and biological properties, the molecular shape largely determines the final function, and this is the rationale for the development of a large number of conformation search methods. The basic approach is to search the conformation space of a molecule in order to find energetically accessible regions. The problem can be broken into two major parts: defining the energy function, and finding efficient methods for performing the conformational search. In this paper we compare the efficiency of two conformation search methods, the genetic algorithm (GA) and parallel direct search (PDS) for performing conformation searches of small to moderately large molecules.

In general, one can decompose the search into two phases. In the first phase, we are interested in getting the lay of the land, by performing a coarse but broad search. This stage generates a number of interesting conformations that can be used as starting guesses for the second phase, which is local energy minimization. The global search phase is conceptually the harder of the two because the size of the space is so large. Additionally, local information about the surface rarely provides definitive clues regarding the location of the global minimum. Because it is difficult to exhaustively search the conformation space of any but the smallest molecules, a number of statistical heuristic methods have been developed [1, 2]. These include pure random search, simulated annealing [3], Cartesian coordinate directed tweak [4], taboo search [5], parallel stochastic methods as in [6, 7] and genetic algorithms [8, 9]. Direct search methods have also been reported [10, 11, 12]. Non-stochastic methods have been developed, including Scheraga's diffusion method [13], and a class of branch-and-bound methods due to Floudas [14]. All of these methods can be made to work well on a selected set of molecules, but it is important to perform head-to-head tests between different methods to assess their relative strengths.

We are primarily concerned here with the efficient broad search of conformation space to generate a set of low energy conformations that can be subjected to local gradient mini-

mization. To this end, we present a comparison of the GA and PDS methods. GAs draw on a set of evolutionary metaphors including selection of fit individuals, mutation, and genetic crossover. PDS methods belong to a class of optimization algorithms developed by Dennis and Torczon [23] that can be viewed as multidirectional line search methods. These methods are robust, simple to implement, and easily parallelized. Our test problems have been gathered from a large standard suite of molecules that are being used to compare other conformational analysis methods [9, 15]. In particular, we are interested in the efficiency of both of these methods as the size of the molecules increases. The energies we compare against are those of the known crystal structures after they have been relaxed using a molecular mechanics force field. A related issue addressed in this paper is the prediction of the minimum amount of computer time (measured in number of energy function evaluations) that is necessary to reliably generate a set of starting points that will yield low energy conformations after gradient minimization.

The paper is organized as follows. Section 2 gives an outline of the GA and PDS algorithms. In Section 3 we describe the test problems and give numerical results. Section 4 follows with an analysis of the results.

2. Computational Methods

2.1. Genetic Algorithms

We present here a brief introduction to our variant of the standard GA method [16]. The most important idea is that we work with a population of "individuals" that will interact through genetic operators to carry out an optimization process. An individual is specified by a *chromosome* that is a bit string of length N_c that can be decoded to give a set of physical parameters. In what follows, chromosome and bit string are synonymous. A fitness function, that is the function to be optimized, is used to rank the individual chromosomes. Optimization proceeds by generating populations whose individuals have increasingly higher fitness. An initial population of N_{pop} individuals is formed by choosing N_{pop} bit strings at

random, and evaluating each individual's fitness.

Subsequent generations are formed as follows. All parents are ranked by fitness and the highest fitness individual is placed directly into the next generation with no change. This is referred to as "elitism". Next, the breeding population is formed, consisting of the top ranked 40% of the population. Pairs of individuals (including the highest fitness individuals) are randomly chosen out of the breeding pool, and their chromosomes are crossed over to form chromosomes for enough individuals to fill the new population. Crossover consists of taking some subset of the bits from parent 1 and the complimentary set of bits from parent 2 and combining them to form the chromosome of child 1. The remaining bits from the two parents are combined to form the chromosome of child 2. Additionally, during replication there is a small probability of a bit flip or mutation in a chromosome. This serves primarily to maintain diversity and prevent premature convergence that occurs when a single very fit individual takes over the entire population early in the evolutionary process. To bound the magnitude of the effect of mutations, the binary chromosomes are Gray coded. An integer that is represented as a Gray coded binary number has the property that most single bit flips change the value of the integer by ± 1.

Our GA code is implemented as a module of CCEMD [17] a general purpose molecular dynamics code that uses the QUANTA/CHARMm [18, 19] force field. The three principal operators are selection of parents, mutation, and crossover. Boxcar selection is used, in which every individual in the top-ranked 40% of the population has an equal chance of being selected for mating. The fitness is the negative of the potential energy. The crossover operator takes pairs of parents and joins each of their binary chromosomes end-to-end to form two rings. It then chooses two symmetric points to cut the rings and forms a pair of children by swapping pieces of the parents' chromosomes. Finally the mutation operator acts by flipping bits in the binary chromosome. Each bit has a probability equal to R_m of being flipped from 1 to 0 or vice versa. Mutation rates are typically quite low, on the order of 0.04. An important detail is that the entire population is not regenerated at each generation. The top 10% of the old population is moved into the new population and all but the single

best are subjected to the mutation operator. We always use the "elitist" strategy in which the most fit individual in each generation is passed directly to the next without crossover or mutation. This ensures that the best individual is never lost, but continues to be available for mating. Note that this individual is transferred directly from generation i to $i+1$ but also produces offspring that make up part of generation $i+1$.

We have the ability to run multiple sub-populations simultaneously. At periodic intervals, these populations can communicate by passing the best individual from each population to each of the others.

During the crossover operations, a niching operation is used. As prospective new members of the population are created, they are compared to those already accepted, by measuring the Hamming distance. The Hamming distance is the fraction of bit positions that have different values in the two chromosomes. The prospective new member is rejected if it is too similar to ones already present. Initially, an individual must differ by 40% from every other individual (i.e. no more than 60% of the bits in the two can by set the same.) As the population fills up, this criteria becomes too restrictive and it is slowly relaxed until the population is filled.

2.2. Direct Search Methods

Direct search methods belong to a class of optimization methods that do not compute derivatives. Examples of direct search methods are the Nelder-Mead Simplex method [20], Hooke and Jeeves' pattern search [21], the box method [22], and Dennis and Torczon's parallel direct search algorithm (PDS) [23]. The PDS algorithm can be viewed as an intelligent adaptive grid search algorithm employing a multi-sided simplex.

Starting from an initial simplex S_o, the function value at each of the vertices in S_o is computed and the vertex corresponding to the lowest function value, v_o, is determined. Using the underlying grid structure, the simplex S_o is rotated 180° about v_o and the function values at the vertices of this rotation simplex, S_r, are compared against v_o. If one of the vertices in the simplex S_r has a function value less than the function value corresponding to v_o, then an

expansion step to form a new simplex, S_e, is attempted in which the size of S_r is expanded by some multiple, usually 2. The function values at the vertices of S_e are compared against the lowest function value found in S_r. If a lower function value is encountered, then S_e is accepted as the starting simplex for the next iteration; otherwise S_r is accepted for the next iteration. If no function value lower than the one corresponding to v_o is found in S_r, then a contraction simplex is created by reducing the size of S_o by some multiple, usually 1/2, and is accepted for the next iteration.

Because PDS only uses function comparisons, it is easy to implement and use. Since the rotation, expansion, and contraction steps are all well-determined it is possible to determine ahead of time a set of grid points corresponding to the vertices of the simplices constructed from various combinations of rotations, expansions, and contractions. Given this set of grid points, called a search scheme, the PDS algorithm can compute the function values at all of these vertices in parallel and take the vertex corresponding to the lowest function value. An interesting consequence of this approach is that the PDS algorithm can jump out of local wells by using a large enough search scheme size. By varying the size of the search scheme one can therefore use the PDS algorithm as a means of efficiently generating conformations in a manner similar to GA and simulated annealing.

It is also worthwhile to contrast PDS with grid search methods. In a grid search method the grids are generated by starting with a fixed molecule and systematically varying one of the internal variables. This method works well for small molecules but becomes computationally prohibitive for larger molecules. The grid in PDS however is adaptive and will automatically change in response to the contours of the energy surface. The PDS code we used is a modification of that developed by Torczon and obtained from the Center for Research on Parallel Computation at Rice University (available via email from softlib@cs.rice.edu).

3. Numerical Results

3.1. Test Problems

In two earlier studies [8, 12], both GAs and simulated annealing outperformed random search for large molecules. Additionally, GA has been shown to outperform simulated annealing in several conformation search applications [8, 24, 25, 26, 27]. GA has been compared against the CSEARCH method [28] and was shown to be more efficient for problems with more than about 8 rotatable bonds. Clark, et al. [24] have compared GA with several other methods and found it superior to all but a directed tweak method [29], to which it performed in a comparable manner. In this section we present the results of numerical experiments comparing PDS with GA.

We tested the two methods on 19 different molecules. The first 10 are taken from a standard set [30] of 72 molecules chosen from the Cambridge Structural Database (CSD) [31]. These range in size from 2 to 12 dihedrals. The structures of the complete set can be found in references [15] and [30]. A set of 7 intermediate size molecules (14-24 dihedrals) was added to the original set. These were also taken from the Cambridge database. Their structures are given in Figures 1 and 2. Finally two peptides were added to the set. These are inhibitors of the S-protein [32] and have sequences CH_3CO-Tyr-Asn-Phe-Glu-Val-Leu-$NHCH_3$ (denoted *sprot* with 32 dihedrals) and CH_3CO-Tyr-Asn-Phe-Glu-Val-Leu-Gly-Lys (denoted *sprotbpcap*1, with 39 dihedrals).

For all but the two peptides, we relaxed the crystal structures using CCEMD and the CHARMm force field with infinite cutoffs, standard charges provided by QUANTA [18], and no dielectric screening. This provided the reference energy which was the target for the optimization methods. The minimized crystal structure was used as the template for the GA. The GA completely randomizes the initial set of dihedrals, so it is not biased towards the reference. Only non-ring dihedrals were varied, so that all rings assumed their relaxed crystal conformations.

To obtain a reference energy for the peptides *sprot* and *sprotbpcap*1, extended structures

were built in QUANTA, and 1 nsec dynamics trajectories were run. The force field was the same as used for the other molecules except for the addition of a continuum solvent term. CCEMD uses a variant on the continuum model of Still, et al. [33]. A number of snapshots were taken along the trajectory and minimized. The lowest energy conformation found became the reference energy.

3.2. GA Parameters

The important variables in the GA method are the population size, the total number of evaluations allowed, the number of bits used to represent a real variable, and the mutation rate. The populations and number of generations were chosen as follows. Four runs were performed for each of the two conformation search methods. These runs used a total of 1000, 5000, 10000, and 40000 potential energy evaluations respectively. For the runs with a total of 1000 function evaluations, the population size was 50 and 20 generations were run. For the 5000 function evaluation case, the population size was 100 and 50 generations were run. For the 10000 function evaluation case, two parameters sets were used. First, one sub-population was run with size 100 for 100 generations. Then a 4 sub-population run was performed with populations of 50 that were run for 50 generations. The populations interacted at generations 20 and 40. The rationale behind trying two sets of this size is to test the notion that isolating the sub-populations will slow convergence, and allow a wider search. For the 40000 function evaluation case, 4 sub-populations were used, each of size 100, run for 100 generations. They interacted at generations 20, 40, 60, and 80. We used 10 bits to represent each angle, which is equivalent to torsion angle resolution of about 0.35°. The mutation rate we used is $R_m = 0.04$. At the end of the search phase for each run, all of the conformations found were clustered, so that any pair of conformations within 40° torsional rms of one another were in the same cluster. Then the lowest energy conformation in each cluster, as well as the overall lowest energy 10 conformations were gradient minimized.

3.3. PDS Parameters

The important parameters for the PDS algorithm are the stopping tolerance and the size of the search scheme. The stopping tolerance we used was quite large because we are not interested in finding a local minimum but instead in searching conformational space. We ran 4 different sets of experiments, as in the GA runs, each one with a different maximum on the number of function evaluations allowed. Since each PDS run requires very few function evaluations, we ran multiple PDS runs starting from different conformations until the maximum number of function evaluations was exceeded. The 20 best conformations (as determined by the potential energy) from this stage were then given to a local minimization routine that computed the final energy.

We also wanted to explore the effect of two different search scheme sizes for PDS. For the large molecule cases, we ran two different sets of PDS runs, with the search scheme size set to $2 * N_{dih}$ and $4 * N_{dih}$. The major difference between the two runs was that more PDS runs were completed with the smaller search scheme size. In almost all cases the smaller scheme size generated lower final energies than the larger scheme size. The one test problem in which this was not true was the $bu f ti f$ molecule for which the larger scheme size predicted a substantially lower energy.

3.4. Comparison of Results

Tables 1-3 contain the GA and PDS minimized energies for all of the test problems. For each molecule, we give the name, the number of dihedrals, the reference energy, and the results from the GA and PDS runs for the 4 basic runs. This column gives the lowest energy found after the search and gradient minimization phases. For ease of comparison, only one of the 10000 function evaluation GA runs is included, that being the case of 1 sub-population.

Figure 3 shows the difference in energy between the reference value and the lowest energy found by each method for the case where the maximum number of function evaluations was set to 1000. The figure clearly indicates that even for the least expensive case, both GA and PDS can quickly achieve conformations that are close to the reference energy. In Figure 4

we present the same results for the lowest energy conformation over all of the numerical experiments. This figure indicates that GA and PDS perform almost identically over the entire set of test problems. Both methods yield conformations that have energies below the reference energy.

Table 4 gives a comparison between the results of the two search approaches used in the GA for the 10000 function evaluation case. The results for 1 sub-population of 100 individuals, run for 100 generations is given in column GA1. The results for the run with 4 sub-populations of 50 individuals and 50 generations is given in column GA2. If one method outperformed the other by more than 1 kcal/mol, it is indicated by a star. For the small half of the molecules, neither method outperformed the other, while for the larger half of the molecules, the two methods found different best local minima, but with roughly equal probabilities of finding the lower energy. From this test, at least, no preference can be given to one approach or the other.

Another question of interest was what effect the search scheme size (SSS) parameter for PDS would have on the conformational search. Since the SSS parameter directly affects the size of the grid on which PDS searches for minima, the larger this parameter is the greater the probability of stepping out of a local minimum well. However, larger values of the SSS parameter also result in fewer starting points being considered. In Table 5 we present the results of running two PDS searches with SSS equal to $2 * N_{dih}$ and $4 * N_{dih}$, for the largest 4 molecules. The results indicate that for 3 out of the 4 molecules a larger value of SSS did not improve the best energy found. The one exception was the *buftif* molecule for which PDS not only found a conformation with a substantially lower energy than with the smaller SSS, but it was also the lowest energy found by either method.

4. Conclusions

The numerical results show that both methods work well even as the size of the molecule increases. For the small molecule cases, PDS seems to work better than the GA, although the differences are small. For the medium molecule cases, GA quickly finds low energy con-

formations as evidenced by the results for the 1000 function evaluation computer runs. The best energy found however was usually from PDS with the single exception being molecule 14, *bulceq*2.

The case for the larger molecules is slightly more interesting. Both methods always found conformations with energies lower than the reference energies. In the large-molecule cases, GA found a lower energy in 3 out of the 4 test problems. However, in all cases there was never more than a 6 kcal/mol difference between the two methods.

As we increase the number of function evaluations allowed, PDS generally tends to find lower energy conformations, but this is not always true. For the GA method, this is even less often the case. It is important to note that the different runs used different initial conformations so that the 5000 function evaluation run was not simply a continuation of the 1000 function evaluation run. If the longer runs *were* just a continuation, then both methods are guaranteed to get successively better results as more function evaluations are used. What this points out is the sensitivity to initial conditions in these and other conformation search methods that have a stochastic component. Although we did not do this test, the numerical results suggest that a GA run using 40 totally independent runs of 1000 evaluations will do better than a single 40000 function evaluation run. From a given starting population, the GA method relatively quickly exhausts its ability to search globally, and instead starts concentrating on small, but promising local regions.

These results suggest that both the GA and PDS methods would be good candidates for use in a screening application. In that case, it is important to quickly generate one or a few low energy conformations for a large (> 1000) set of molecules. For the set of molecules considered here with up to 8 dihedrals, the lowest energy found by any method was achieved within 5000 function evaluations. With our current code, running on an SGI Power Challenge, 5000 function evaluations takes on the order of 1 cpu minute for a typical molecule. For the largest molecule considered here, *sprotbpcap*1, the 40000 function evaluation run took about 1 cpu hour, which includes extra overhead to calculate the continuum solvation term.

Acknowledgements

The authors acknowledge partial support by the Department of Energy under contract DE-ACO4-94AL85000 with Sandia National Laboratories.

REFERENCES

[1] A.E. Howard and P.A. Kollman. *J. Med. Chem.*, 31:1669–1675, 1988.

[2] Martin Saunders, K.N. Houk, Yun-Dong Wu, W. Clark Still, Mark Lipton, George Chang, and Wayne C. Guida. *J. Am. Chem. Soc.*, 112:1419–1427, 1990.

[3] S.R. Wilson and W. Cui. *Biopolymers*, 29:225–235, 1990.

[4] M. Saunders. *J.Comp.Chem.*, 20:203, 1989.

[5] D. Cvijovic and J. Klinowski. *Science*, 267:664, 1995.

[6] R.H. Byrd, E. Eskow, R.B. Schnabel, and S.L. Smith. Technical Report CU-CS-553-91, University of Colorado at Boulder, 1991.

[7] R.H. Byrd, E. Eskow, and R.B. Schnabel. Technical Report CU-CS-630-92, University of Colorado at Boulder, 1992.

[8] R.S Judson, M.E. Colvin, J.C. Meza, A. Huffer, and D. Gutierrez. *International Journal of Quantum Chemistry*, 44:277–290, 1992.

[9] R.S Judson, E.P. Jaeger, A.M. Treasurywala, and M.L. Peterson. *J.Comp.Chem.*, 14:1407, 1993.

[10] R.S. Maier, J.B. Rosen, and G.L. Xue. Technical Report 92-031, AHPCRC, 1992.

[11] G.L. Xue. Technical Report 92-055, University of Minnesota, 1992.

[12] J.C. Meza and M.L. Martinez. *Journal of Computational Chemistry*, 15(6):627–632, 1994.

[13] J. Kostrowicki, L. Piela, B. Cherayil, and H.A. Scheraga. *J.Phys.Chem.*, 95:4113, 1991.

[14] C.D. Maranas and C.A. Floudas. *J.Chem.Phys.*, 100:1247, 1994.

[15] A.K. Ghose, E.P. Jaeger, P.J. Kowalczyk, M.L. Peterson, and A.M. Treasurywala. *J. Comp. Chem.*, 14:1050, 1993.

[16] D.E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning.* Addison-Wesley, 1989.

[17] R.S. Judson, A. Windemuth, D.B. McGarrah, C.F. Melius, E. Mori, D. Barsky, Y.T. Tan, A.M. Treasurywala, E.P. Jaeger, J.C. Meza, and T. Plantenga. CCEMD, Version 2.2, Sandia National Laboratories, 1995.

[18] 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.

[19] B.R. Brooks, B. Bruccoleri, B.D. Olafson, D.J. States, S. Swaminathan, and M. Karplus. *J.Comp.Chem.*, 4:187–217, 1983.

[20] J. Nelder and R. Mead. *Comput. J.*, 7:308–313, 1965.

[21] R. Hooke and T.A. Jeeves. *J. Assoc. Comp. Mach.*, 8:212–229, 1961.

[22] G.E.P. Box and K.B. Wilson. *J. Royal Statistical Society, Series B*, 13:1–45, 1951.

[23] J.E. Dennis and V.J. Torczon. *SIAM J. Optimization*, 1(4):448–474, 1991.

[24] D.E. Clark, G. Jones, P. Willett, P.W. Kenny, and R.C. Glen. *J.Chem.Inf.Comput.Sci.*, 34:197, 1994.

[25] P. Tuffery, C. Etchebest, S. Hazout, and R. Lavery. *J.Comp.Chem.*, 14:790, 1993.

[26] R. Unger and J. Moult. *J.Mol.Bio.*, 231:75, 1993.

[27] J.R. Gunn, A. Monge, R.A. Friesner, and C.H. Marshall. *J.Phys.Chem.*, 98:702, 1994.

[28] R.A. Dammkoehler, S.F. Karasek, E.F.B. Shands, and G.R. Marshall. *J.Comp.-Aided Mol. Design*, 3:3, 1989.

[29] P.S. Shenkin, D.L. Yarmush, R.M. Fine, H. Wang, and C. Levinthal. *Biopolymers*, 26:2053, 1987.

[30] M. Clark, R.D. Cramer, and N.V. Opdenbosch. *J. Comp. Chem.*, 10:982, 1989.

[31] F.H. Allen, S.A. Bellard, M.D. Brice, B.A. Cartwright, A. Doubleday, H. Higgs, T. Hummelink, B.G. Hummelink-Peters, O. Kennard, W.D.S. Motherwell, J.R. Rodgers, and D.G. Watson. *Acta Crystallogr*, 35, Sect. B:2331, 1979.

[32] P.Y. Huang, and R. Carbonell. Biotechnology and Bioengineering (in press).

[33] W. Hasel, T.F. Hendrickson, and W.C. Still. *Tetrahedron Comp.Meth.*, 1:103, 1988.

Table 1: Summary of Results: small molecules.

No.	Molecule	N_{dih}	Reference	FCN evals	GA	PDS
1	*abaxes*	2	0.5	1000	0.5	0.5
				5000	0.5	0.5
				10000	0.5	0.5
				40000	0.5	0.5
2	*acfpch*	3	2.3	1000	2.3	2.2
				5000	2.2	2.2
				10000	2.2	2.2
				40000	2.3	2.2
3	*abinor2*	4	15.0	1000	9.2	6.8
				5000	9.2	9.2
				10000	9.2	9.2
				40000	9.2	9.2
4	*acdxur*	5	-2.7	1000	-5.2	-7.8
				5000	-6.0	-6.1
				10000	-6.1	-6.1
				40000	-6.0	-6.1
5	*abtoet*	6	21.2	1000	17.5	12.8
				5000	17.5	12.8
				10000	17.5	17.4
				40000	17.5	17.4
6	*acthbz*	6	11.4	1000	4.5	4.8
				5000	4.5	4.4
				10000	4.5	4.4
				40000	4.5	4.4
7	*acarap*	8	-95.7	1000	-98.4	-99.9
				5000	-98.4	-99.9
				10000	-98.4	-99.9
				40000	-99.8	-98.4

Table 2: Summary of Results: medium molecules.

No.	Molecule	N_{dih}	Reference	FCN evals	GA	PDS
8	*aaxthp*	10	-97.2	1000	-101.9	-100.1
				5000	-102.5	-102.5
				10000	-102.6	-102.7
				40000	-102.6	-102.7
9	*acinst*	10	-87.4	1000	-90.7	-80.9
				5000	-87.3	-87.4
				10000	-87.3	-90.8
				40000	-87.4	-90.8
10	*acbuol*	12	-10.5	1000	-14.2	-13.4
				5000	-15.3	-15.5
				10000	-14.4	-16.6
				40000	-14.4	-17.5
11	*cazkuj*	13	31.1	1000	23.9	14.9
				5000	17.3	14.9
				10000	20.7	14.9
				40000	20.5	21.5
12	*cacsii*	14	168.8	1000	169.9	171.9
				5000	167.9	168.0
				10000	167.5	167.1
				40000	166.4	164.1
13	*bettez*	16	181.7	1000	178.7	181.0
				5000	179.6	174.0
				10000	179.2	174.0
				40000	178.8	174.0
14	*bulceq*2	18	-77.4	1000	-75.8	-73.3
				5000	-79.6	-77.3
				10000	-79.7	-77.3
				40000	-81.6	-76.5
15	*bagzue*4	19	35.0	1000	25.7	25.0
				5000	29.8	23.2
				10000	28.1	25.0
				40000	25.0	27.4

Table 3: Summary of Results: large molecules.

No.	Molecule	N_{dih}	Reference	FCN evals	GA	PDS
16	*bedgew*	22	-73.4	1000	-84.8	-79.2
				5000	-74.0	-79.8
				10000	-95.8	-89.0
				40000	-88.1	-85.9
17	*buftif*	24	-205.1	1000	-194.2	-183.9
				5000	-206.4	-191.5
				10000	-212.2	-191.5
				40000	-208.3	-193.0
18	*sprot*	32	-354.0	1000	-342.4	-337.4
				5000	-362.0	-352.5
				10000	-348.9	-352.5
				40000	-360.7	-357.8
19	*sprotbpcap*1	39	-558.0	1000	-563.4	-534.6
				5000	-538.1	-541.4
				10000	-541.8	-543.9
				40000	-566.7	-567.9

Table 4: Effect of the number of niches on final energy, 10000 functions. A * indicates which, if either, of the two runs found an energy at least 1 kcal/mol lower than the other.

No.	Molecule	N_{dih}	Reference	GA1	GA2
1	*abaxes*	2	0.5	0.5	0.5
2	*acfpch*	3	2.3	2.3	2.2
3	*abinor2*	4	15.0	9.2	9.2
4	*acdxur*	5	-2.7	-6.1	-6.0
5	*abtoet*	6	21.2	17.5	17.5
6	*acthbz*	6	11.4	4.5	4.5
7	*acarap*	8	-95.7	-98.4	-99.9*
8	*aaxthp*	10	-97.2	-102.6	-102.6
9	*acinst*	10	-87.4	-87.3	-87.3
10	*acbuol*	12	-10.5	-14.4	-14.4
11	*cazkuj*	13	31.1	20.7	19.5*
12	*cacsii*	14	168.8	167.5	166.4*
13	*bettez*	16	181.7	179.2*	181.4
14	*bulceq2*	18	-77.4	-79.7	-79.8
15	*bagzue4*	19	35.0	28.1	17.3*
16	*bedgew*	22	-73.4	-95.8*	-82.4
17	*buftif*	24	-205.1	-212.2*	-189.8
18	*sprot*	32	-354.0	-348.9	-352.8*
19	*sprotbpcap1*	39	-558.0	-541.8*	-537.6

Table 5: Effect of search scheme size (SSS) on final energy: large molecules.

No.	Molecule	N_{dih}	Reference	FCN evals	SSS=2N	SSS=4N
16	*bedgew*	22	-73.37	1000	-79.2	-72.2
				5000	-79.8	-80.0
				10000	-89.0	-80.0
				40000	-85.9	-85.9
17	*buftif*	24	-205.07	1000	-183.9	-194.6
				5000	-191.5	-195.3
				10000	-191.5	-196.8
				40000	-193.0	-209.5
18	*sprot*	32	-354.0	1000	-337.4	-337.4
				5000	-352.5	-337.4
				10000	-352.5	-337.4
				40000	-357.8	-352.5
19	*sprotbpcap*1	39	-558.0	1000	-534.6	-524.4
				5000	-541.4	-542.1
				10000	-543.9	-542.1
				40000	-567.9	-542.1

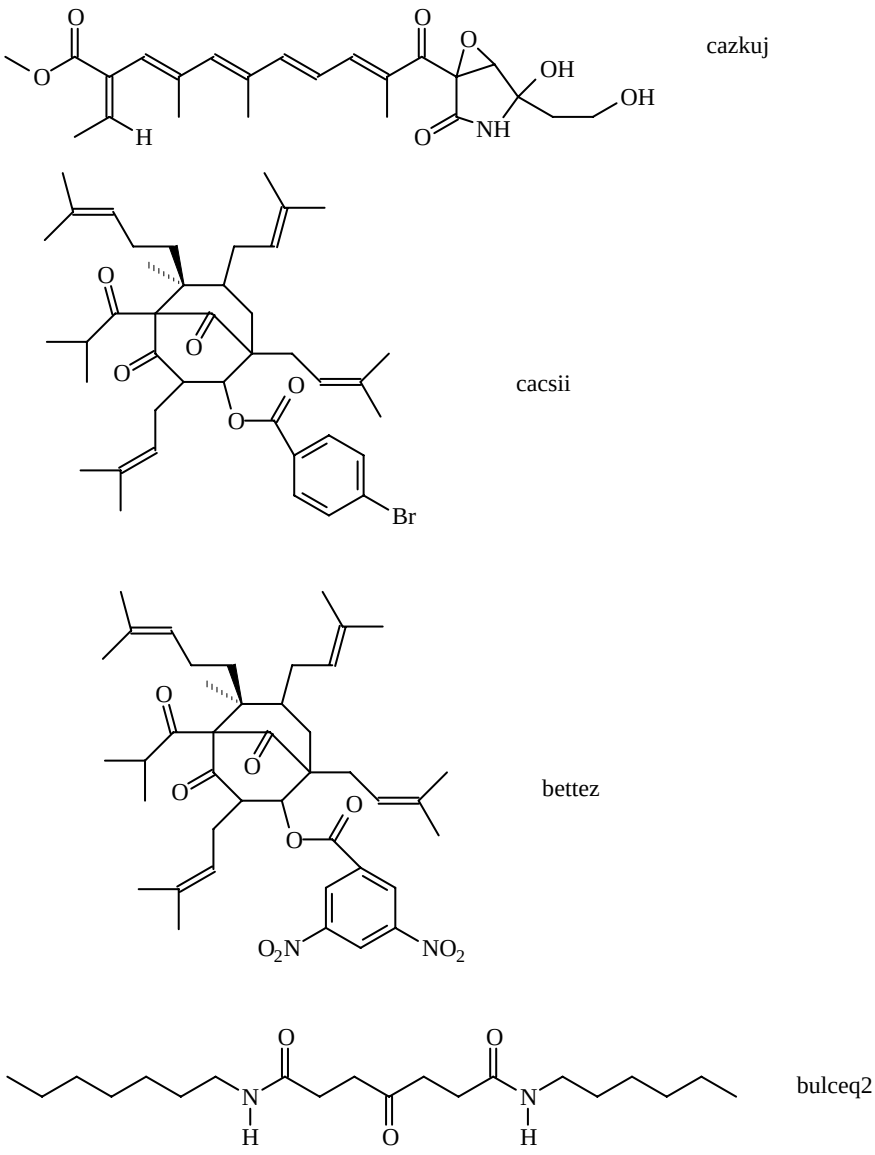

Figure 1: Intermediate and large molecules

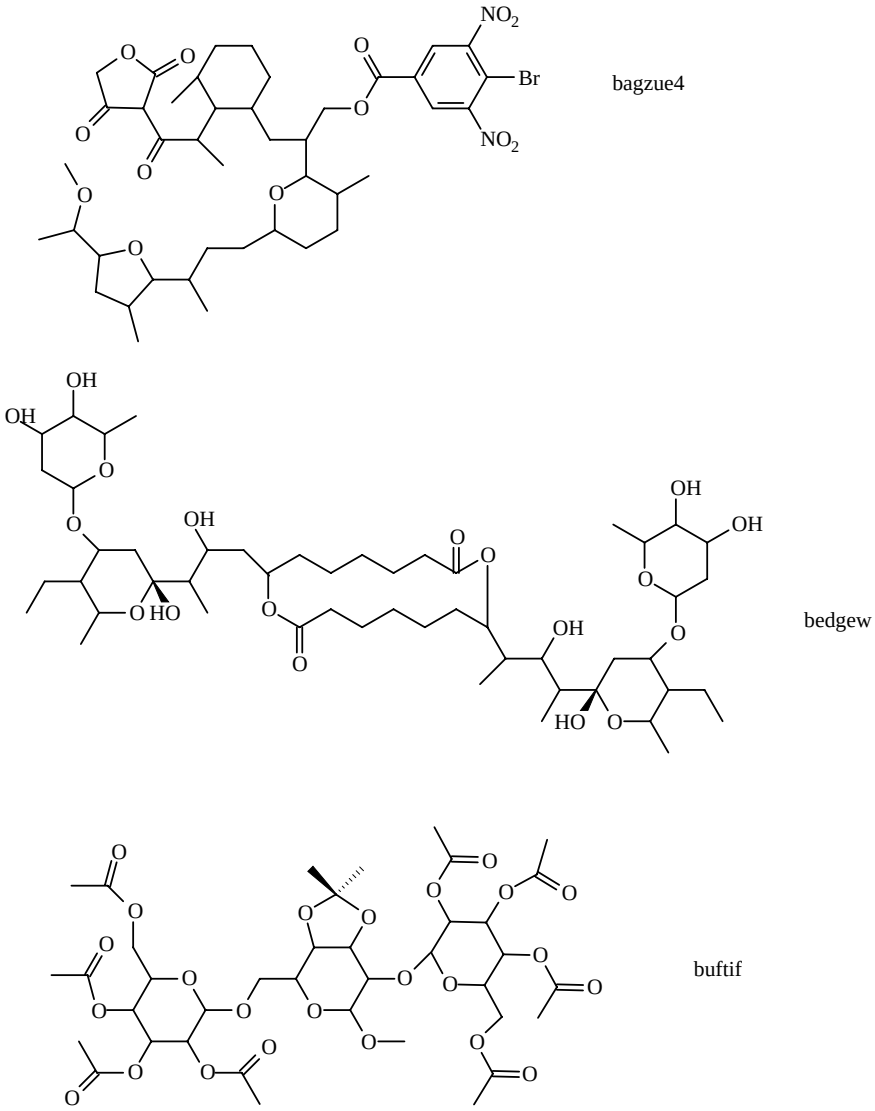

Figure 2: Continuation

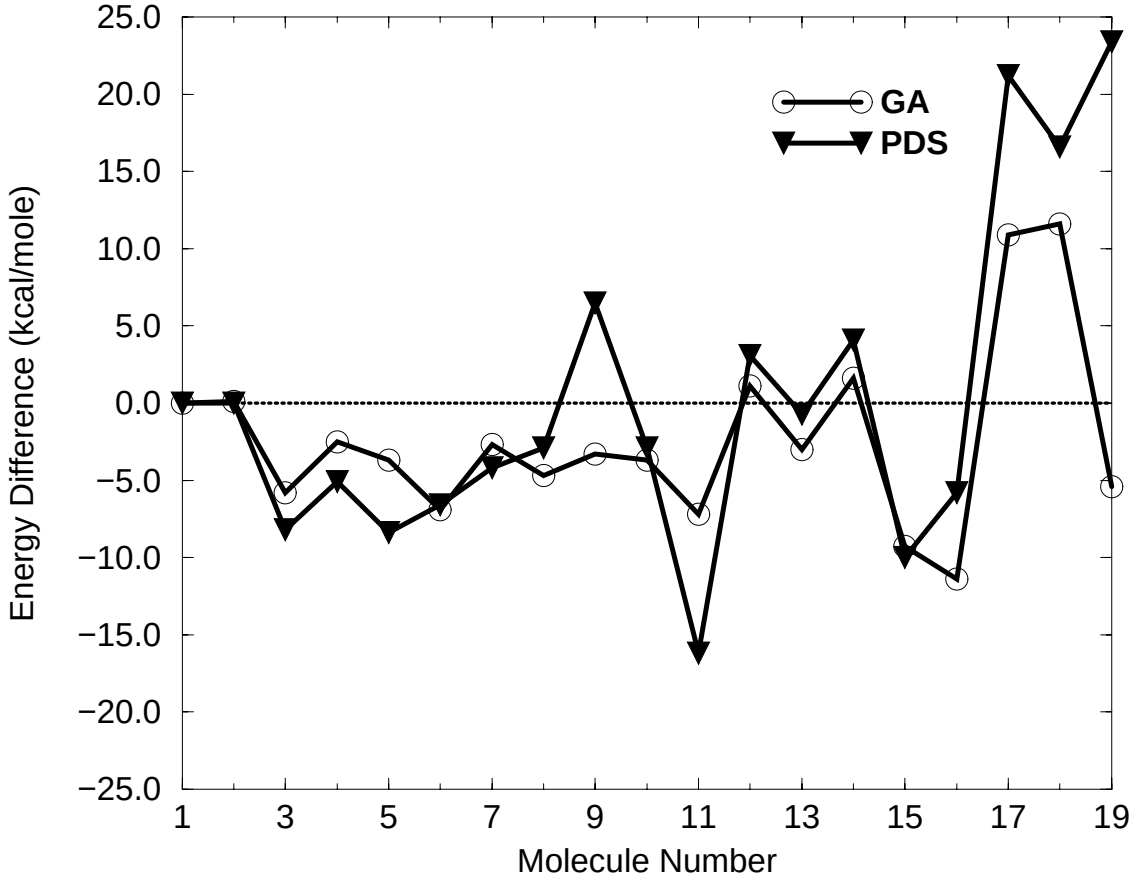

Figure 3: Energy difference between the reference value and the lowest energy found after 1000 function evaluations. The molecule number refers to that given in Table 1.

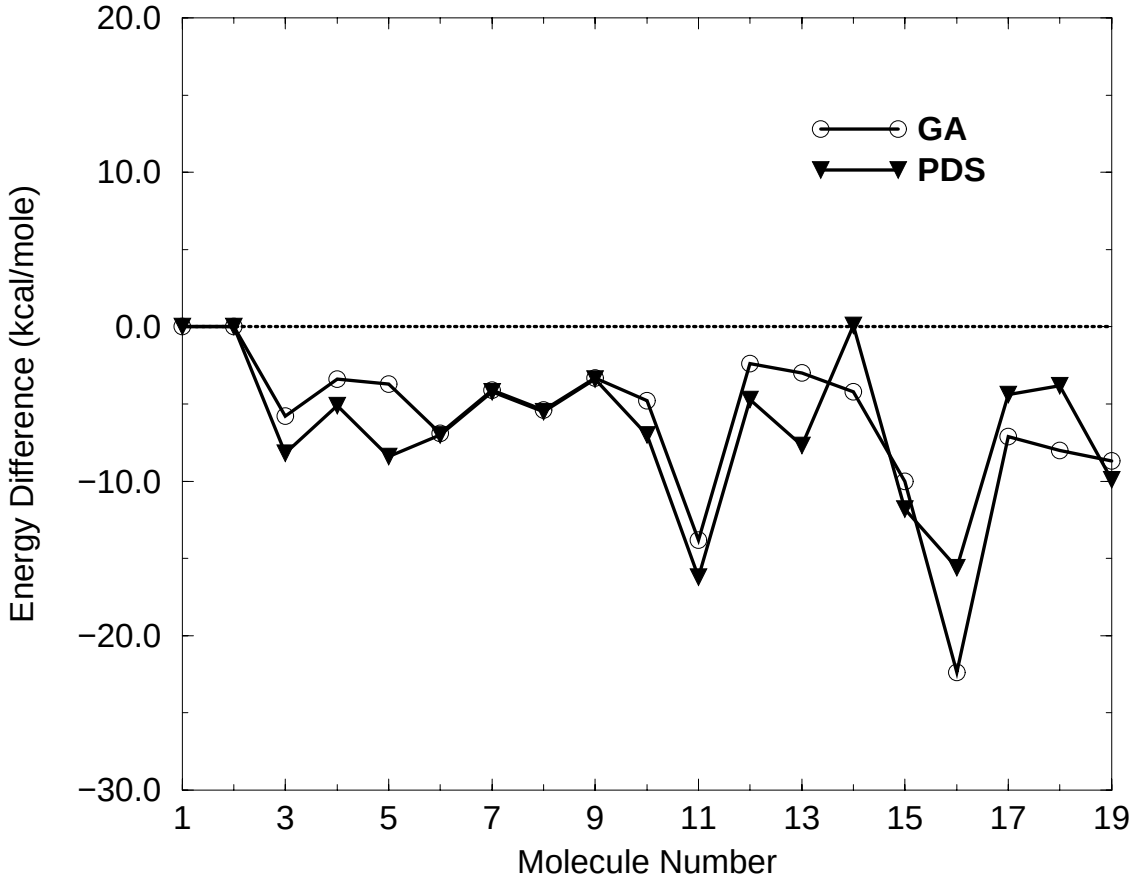

Figure 4: Energy difference from reference energy for best conformation found from any of the runs.