A Hybrid Genetic Algorithm with Pattern Search for Finding Heavy Atoms in Protein Crystals

Joshua L. Payne
Dept. of Computer Science
University of Vermont
Burlington, VT 05405
802-656-3330
Joshua.Payne@uvm.edu

Margaret J. Eppstein
Dept. of Computer Science
University of Vermont
Burlington, VT 05405
802-656-1918
Maggie.Eppstein@uvm.edu

ABSTRACT
One approach for determining the molecular structure of proteins is a technique called iso-morphous replacement, in which crystallographers dope protein crystals with heavy atoms, such as mercury or platinum. By comparing measured amplitudes of diffracted x-rays through protein crystals with and without the heavy atoms, the locations of the heavy atoms can be estimated. Once the locations of the heavy atoms are known, the phases of the diffracted x-rays through the protein crystal can be estimated, which in turn enables the structure of the protein to be estimated. Unfortunately, the key step in this process is the estimation of the locations of the heavy atoms, and this is a multi-modal, non-linear inverse problem. We report results of a pilot study that show that a 2-stage hybrid algorithm, using a stochastic genetic algorithm for stage 1 followed by a deterministic pattern search algorithm for stage 2, can successfully locate up to 5 heavy atoms in computer simulated crystals using noise free data. We conclude that the method may be a viable approach for finding heavy atoms in protein crystals, and suggest ways in which the approach can be scaled up to larger problems.

Categories and Subject Descriptors
J.3 Computer Applications [**Life and Medical Sciences**]: *biology*

General Terms
Algorithms, Performance, Experimentation.

Keywords
Crystallography, crystallographic phasing, phase problem, isomorphous replacement, heavy atom method, hybrid evolutionary algorithms, genetic algorithms, pattern search.

GECCO'05, June 25-29, 2005, Washington, DC, USA.

1. INTRODUCTION
Our knowledge of molecular structures of proteins at atomic resolution comes largely from an imaging process known as x-ray crystallography [16],[17]. Using x-rays in much the same way as light is used in a conventional microscope, this method differs from ordinary microscopy in that there are no lenses for x-rays. Instead, the scattered (diffracted) x-rays from the specimen are measured, and the functionality of the lens is mimicked by the Fourier transform. Unfortunately, an important property of the scattered x-rays, their individual phases, cannot be measured by any detector. Only the intensities are measured, the inverse Fourier transform of which gives a 3D image (the Patterson map) representing the superposition of all the inter-atomic vectors of the molecule. The Patterson map can be thought of as the convolution of the molecular image with itself. For biologically-interesting molecules like enzymes and other proteins comprised of several thousand atoms, the immense number of overlapping inter-atomic vectors precludes deconvolution of the Patterson map.

To solve this problem, crystallographers often use an approach known as *iso-morphous replacement* [16],[17], in which they 'dope' their protein specimens with heavy atoms like mercury and platinum, whose inter-atomic vectors are much stronger than the vectors arising from the much lighter atoms comprising proteins. Determining exactly where these heavy atoms bind to the protein is the first step in determining the protein's structure. In brief, the heavy atoms perturb the intensities of the scattered rays in a way that allows the phases of the protein's contribution to the scattering to be determined relative to the phases of the heavy atoms' contribution. The latter phases can be derived directly once the locations of the heavy atoms are known. The difficulty of the entire imaging procedure often lies in locating the heavy atoms, a problem that has a multi-modal fitness landscape that is not linearly decomposable. Numerous computational algorithms have been developed which address this problem, some of which are reasonably successful [3],[6],[18],[19]. However, cases still arise in which these heavy-atom search programs fail.

Buoyed by the recent successes of evolutionary algorithms applied to various crystallographic problems [1],[2],[9],[13], we decided to explore their application to the task of locating heavy atoms within crystals. In particular, we investigated the use of a genetic algorithm (GA) for identifying one promising solution in the neighborhood of one of the many global optima (a non-deterministic exploration phase), then subsequently refined the

estimated solution with a local hill-climbing algorithm (a deterministic exploitation phase). This back-to-back hybridization scheme differs from memetic approaches [14] that apply a local search to multiple individuals in the population in between generations of a GA. Similar tandem hybridization of global and local search algorithms have been shown to be successful in other difficult optimization problems [15],[20],[23]. For the local search algorithm, we selected a deterministic pattern search (PS) algorithm [10]. Unlike in [7],[8], where a PS is used in a hybrid algorithm to govern the mutation step size and direction of an evolutionary algorithm, herein we hybridize the GA and PS in a back-to-back manner, as suggested (but not implemented) by [21].

In order to more accurately assess the effectiveness of our hybrid algorithm as a search technique capable of navigating the difficult multi-modal landscape of the heavy atom location problem, we performed a pilot study using noise-free computer-generated data that permitted us to use a fairly simple fitness function. This is not intended to minimize the importance or difficulty of constructing a viable fitness function that is robust in the face of noisy crystallographic data.

2. METHODS

2.1 Creation of Synthetic Test Data

Protein crystals comprise many repetitions of parallelepipids, called *unit cells*, arranged on a crystal lattice. Each unit cell contains extensive internal symmetry, and crystals are divided into 230 *space groups*, defined by the type of symmetry that extends throughout the crystal. All atom locations in a crystal are uniquely defined by the locations of atoms in a smaller *asymmetric unit* (AU); applying symmetry operators for the crystal's space group to the atoms in the AU generates all atom locations within the unit cell (e.g., see Table 1). Coordinates in the unit cell are normalized to the range [0,1] in each of the 3 space dimensions. For any crystal, there are multiple equivalently valid options for locating the origin of the AU.

Table 1. For each atom located at <x,y,z> in the AU, symmetry in the P422 space group results in the following 8 atoms in the unit cell.

x_j	y_j	z_j
x	y	z
-x	-y	z
-x	y	-z
x	-y	-z
-y	-x	-z
y	x	-z
-y	x	z
y	-x	z

Consider a crystal where the j^{th} atom in the unit cell is located at position $<x_j,y_j,z_j>$. The complex *structure factors* F_{hkl}, for each *hkl* reflection of x-rays, are calculated using a Fourier transform, as follows [16]:

$$F_{hkl} = \sum_j f_j \exp\left(2\pi i\left(hx_j + ky_j + lz_j\right)\right)\exp\left(-\frac{B_j}{2}\sqrt{\frac{h^2}{a^2}+\frac{k^2}{b^2}+\frac{l^2}{c^2}}\right) \quad (1)$$

Where *h,k,l* are the indices of the reflection planes in the *x,y,z* dimensions; *a,b,c* are the dimensions of the unit cell in angstroms; f_j is the number of electrons in atom *j*; and B_j is the temperature factor that accounts for the effect of thermal vibration of the atom. For simplicity, equation (1) assumes right-rectangular unit cells, although this assumption is easily relaxed [16]. Equation (1) was implemented using a discrete Fourier transform.

For the purposes of this study we computer-generated noise-free data from several randomly generated synthetic crystals in the space group P422. Although the AU in P422 has coordinates in the range [0,0.5], our synthetic crystals had atom locations in the AU randomly generated within the range [0.15,0.35]. This restriction precludes the need for complicated wrap-around effects and weightings at the edges of the AU and therefore simplified the computations. For each atom in the AU, atom locations in the unit cell were subsequently generated by the P422 symmetry operators (Table 1). All simulated crystals in this pilot study were cubic with *a=b=c=20* angstroms, and were sampled by *512 hkl* reflections (*8* reflection planes in each dimension), corresponding to a 2.5 angstrom sampling resolution, with uniform temperature factor $B_j=10$ angstoms^2, $\forall j$. In a real experiment, crystallographers estimate the *amplitudes* of structure factors due to heavy atoms alone by subtracting amplitudes of reflections from crystals containing protein atoms alone from those due to proteins doped with heavy atoms, thereby introducing additional noise into the heavy atom structure factor estimates. Here, we modeled only heavy atoms (i.e., without protein background) with $f_j=40$ angstoms^2, $\forall j$, and subsequently computed the noise-free amplitudes of the structure factors due to the heavy atoms alone by taking the absolute value of the square of equation (1). Although we could have easily simulated a protein background, this would have made it impossible to separate the effects of noise in the fitness function from the effects of a poor search strategy.

2.2 Fitness function

Because only amplitudes, and not phases, of the structure factors can be measured using x-ray crystallographic techniques, equation (1) is not directly invertible. Our goal was thus to solve the ill-posed inverse problem of estimating $<x_j,y_j,z_j>$, $\forall j=1..H$, given $\left|F_{hkl}^2\right|$, where *H* is the number of heavy atoms in the AU (assumed known). We represented the unknowns in a potential solution with a real-valued vector of locations (**loc**) of length *3H* as follows:

$$\mathbf{loc} = < x_1, y_1, z_1, x_2, y_2, z_2, ..., x_H, y_H, z_H > \quad (2)$$

where x_j,y_j,z_j are bounds-constrained to the range [0,0.5].
By computing the inverse Fourier transform of the amplitudes of the structure factors (with $\left|F_{000}^2\right| = 0.0$), one generates a 3D Patterson map (*P*), as follows [16]:

$$P(u,v,w) = \frac{1}{V}\sum_h\sum_k\sum_l \left|F_{hkl}^2\right| \exp\left(-2\pi i\left(hu+kv+lw\right)\right) \quad (3)$$

where *V* is the volume of the unit cell, and *u,v,w* are the indeces of the 3D Patterson map. The resolution of the Patterson map is

determined by the sampling density of the reflection planes. In these experiments, the Patterson maps were *8×8×8*. Equation (3) was implemented using a fast Fourier transform. A high-level flowchart for computation of the Patterson maps is shown in Figure 1. Patterson maps computed by (3) were further normalized by a) truncating all values more than 2 standard deviations from the mean, and b) dividing by the root mean square of all the map values.

In order to determine the *fitness* of a potential solution, we compute the Pearson's correlation coefficient (*r*) between the normalized Patterson map of the estimated solution and the normalized Patterson map generated from the "true" heavy atom locations, both computed as shown in Figure 1. We define fitness(**loc**) = -*r*, and our search procedures attempt to minimize fitness.

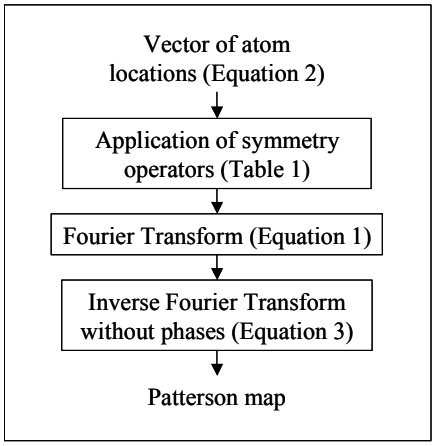

Figure 1: Flowchart for creating a Patterson map given atom locations.

As previously discussed, the Patterson map reflects the inter-atomic distances between all pairs of atoms in the unit cell. Since mirror-image solutions (known as different *hands*) will yield identical Patterson maps, and for each hand there are multiple possible origins for the AU, the mapping from the Patterson image to the constellation of heavy atoms is degenerate [16]. I.e., there are several distinct but equivalent constellations of heavy atom locations in the AU that will give rise to the same Patterson image. This implies that the fitness landscape has multiple global optima. For example, in space group P422 there are 8 global optima. Furthermore, the problem is not linearly separable into locating individual atoms. Although locating one (of several) heavy atoms correctly in the AU will generate 8 *self peaks* that are correctly located in the Patterson map (due to the inter-atomic distances of the 8 symmetry-generated atoms in the unit cell), the vectors between these atoms and any other incorrectly-located atoms (or other correctly-located atoms in a different hand and /or origin) will give rise to *cross peaks* that are incorrectly located in the Patterson map. Computational experimentation showed that, for our synthetically generated crystals, an atom must be within about 2 angstroms of the true atom location for the solution to be able to hill-climb in the fitness landscape, even if all the other atoms have been correctly located.

We implemented the fitness function in the Matlab programming language (V. 7.0) [12], making heavy use of Matlab's vectorizing capabilities to achieve computational efficiency. Our experiments were run on a variety of microcomputers with different clock speeds. The time to compute each fitness evaluation depends on a variety of factors, including the number of heavy atoms, the number of reflections, and the speed of the computer. For reference, on a 2.2 GHz Pentium IV, for a 5 heavy atom problem with 512 reflections, our implementation of the fitness function required about 0.015 seconds. In Section 3 we report the machine-independent number of fitness evaluations required for the various algorithms and problems.

2.3 Search strategies

To search this highly multimodal space, we investigated a two-phase hybrid method we dubbed GAPS (Genetic Algorithm with Pattern Search). The first phase is explorative, employing a traditional GA to identify promising areas of the search space. The best solution found by the genetic algorithm (GA) is then refined using a pattern search (PS) method during a subsequent exploitative phase. In order to ascertain the relative contributions of each of the two phases, we also performed paired experiments using the GA only and the PS only. Each of these algorithms are described below.

2.3.1 Genetic Algorithm

We implemented a genetic algorithm (GA) by customizing the Matlab GADS toolbox V. 1.01 [12] to minimize the fitness function defined in Section 2.2. GA parameters were determined by empirical tuning. We used tournament selection with tournaments of size 2 (tournaments of size 4 caused premature convergence). Each generation, 80% of the population was subjected to uniform crossover (which was found to outperform single-point crossover) and the remaining 20% of the population was subjected to mutation using a bounds-constrained Gaussian mutation operator, drawn from a normal distribution with standard deviation of 0.5. Variables that mutated to values outside the feasible range of [0, 0.5] were repaired by random reset to a uniformly generated value in the feasible range. Elitism ensured that the single best individual survived each generation. Required population size was estimated as a function of the number of heavy atoms in the AU, by running the GA with various problem sizes and population sizes for 10 repetitions of one arbitrarily generated problem of each size, ranging from 1 to 5 heavy atoms, and conservatively determining the minimum population size needed to successfully locate all the heavy atoms in each problem. These tests resulted in population sizes of 100, 300, 600, 1000, and 1500, to solve problems with 1, 2, 3, 4, and 5 heavy atoms in the AU, respectively.

Unfortunately, it is impossible to determine the hand and origin of a given atom in an estimated solution, so niching techniques, such as fitness sharing [5] or crowding [11], are not viable approaches for searching this multi-modal landscape (see Section 4 for more details).

2.3.2 Pattern Search

We implemented a bounds-constrained pattern search (PS) using the Matlab GADS toolbox V. 1.01 [12]. At iteration *k*, the PS algorithm samples the solution space in a regular pattern around a single current solution **loc**, in order to try to find a better solution by minimizing the fitness function described in Section 2.2. We implemented a *6H*-point orthogonal pattern with complete

```
Δ1 = 1.0
k = 1
currentFitness = fitness(loc)
while Δk ≥ 1e-6
   bestFitness = currentFitness
   betterFound = false
   for i = 1, 2, …6*H
      if odd(i) then dir = +1, else dir = -1, endif
      index = ⌊(i+1)/2⌋
      newloc = loc
      newloc(index) = max(min(newloc(index)+dir* Δk,0.5),0)
      trialFitness = fitness(newloc)
      if trialFitness  < bestFitness
          betterFound = true
          bestFitness = trialFitness
          bestLoc = newLoc
      endif
   endfor
   if betterFound
      loc = bestLoc
      currentFitness = bestFitness
      Δk+1 = 2Δk
   else
      Δk+1 = ½ Δk
   endif
    k = k+1
endwhile
```

Figure 2: Pseudo code for PS algorithm, where variables in bold are vectors of the form defined by equation (2), and fitness is being minimized.

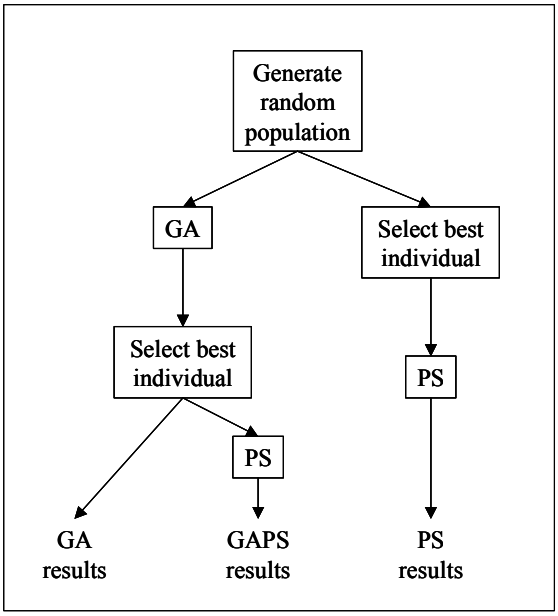

Figure 3: Flowchart for each trial replicate of the GA, PS, and GAPS algorithms.

polling, as follows. For each of the H heavy atoms in the estimate, the PS perturbs the atom by $\pm\Delta_k$, where Δ_k is a real positive step size, along each of the *3* space dimensions. Specifically, $\forall j=1..H$, with current estimate $\langle x_j,y_j,z_j\rangle$, the PS independently samples $\langle x_j\pm\Delta_k,y_j,z_j\rangle$, $\langle x_j,y_j\pm\Delta_k,z_j\rangle$, and $\langle x_j,y_j,z_j\pm\Delta_k\rangle$, keeping all other atom locations unchanged; if a trial step is infeasible, then the sample is simply projected back onto the boundaries of the feasible region. If any of these *6H* perturbations yields a solution with improved fitness, the PS updates the estimate to the best of the sampled solutions in the current iteration and the step size is then doubled for the next iteration; otherwise, the estimate remains unchanged and the step size is halved for the next iteration. The dynamic nature of the step size helps the algorithm to escape local minima [10], much as increases in the "temperature" does in simulated annealing [22]. We set the PS to terminate when $\Delta_k<$*1e-6*. Pseudo code for the PS algorithm is shown in Figure 2.

2.3.3 Genetic Algorithm with Pattern Search

Our two-phase hybrid Genetic Algorithm with Pattern Search (GAPS) was implemented by simply taking the best solution output by the GA and feeding it into the PS algorithm.

2.4 Experimental Design

Experiments were designed to assess the performance of the GA, PS, and GAPS algorithms. Using the method outlined in Section 2.1., we simulated data from a total of 55 random problems (11 random problems for each of 1, 2, 3, 4, and 5 heavy atom problems, designated 1HA, 2HA, … 5HA, respectively). We performed 20 repetitions of each problem for a total of 1100 runs for each of the three methods, using uniformly randomly initialized populations with population sizes described in Section 2.3.1. The random number generator was seeded with the repetition number, so that the GA, PS, and GAPS algorithms started each paired repetition with identical initial populations. For each of the repetitions, we thus followed the flowchart shown in Figure 3.

2.5 Metrics for solution quality

The final fitness of each estimate (i.e., the Patterson correlation coefficient r), is an indirect measure of solution quality. However, since in these synthetically generated problems we know the "true" locations of the heavy atoms, we also assessed our solutions by more direct measures, as follows.

For each repetition of an experiment, we first computed the pair-wise Euclidean distances between each true atom and each estimated atom, for each of the 8 possible hands and origins, and mapped each estimated HA to its closest true HA. We then selected the hand and origin of the true solution for which the sum of the mapped distances was the least. In a real application, the true number of heavy atoms may not be known with certainty. Consequently, in order to assess a) how many heavy atoms were located by our algorithms, and b) how close to the true atom locations the identified heavy atoms were, we did not enforce a one-to-one mapping between true and estimated atoms. In some cases more than one estimated atom mapped to the same true atom, as illustrated in Figure 4 for a hypothetical 5HA estimate. For the selected hand and origin, we report the maximum distance between the estimated atoms and their closest true atoms and the number of true heavy atoms found (i.e., to which an estimated atom had mapped), as shown in Figure 4.

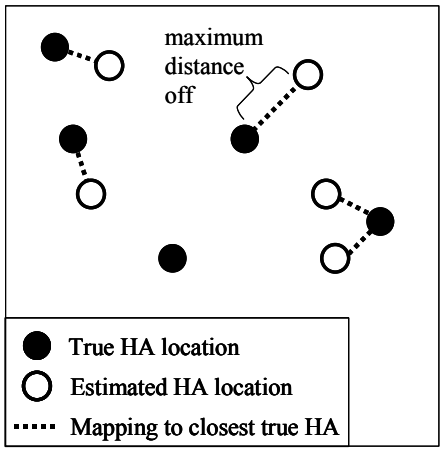

Figure 4: Hypothetical 5HA estimate, with 4 HA found and maximum distance indicated. For clarity, we illustrate this in 2D.

For each method, we averaged the final Patterson correlation *r*, maximum distance, and number of HA found, across all 220 trials for each of the eleven 1HA, 2HA … 5HA problems. Performance metrics were compared across paired replicates between the GA and GAPS, and the PS and GAPS, using a 2-tailed paired Student's t-test. For each algorithm, we also assessed the percentage of trials that found all the heavy atoms and report the average number of fitness function evaluations required.

3. RESULTS

GAPS found solutions with significantly higher Patterson correlations than either the GA or PS methods ($p < 0.005$), for all sizes of problems, as shown in Table 2. Differences in the direct measures of solution quality are even more striking. The GAPS solutions exhibited less spatial deviation from true atom locations than those of the other two methods ($p < 0.0005$), for all problem sizes (Table 3). For example, on the most difficult (5HA) problems, the maximum distance between true and estimated atom locations as determined by GAPS averaged 0.31 angstroms, while the GA alone averaged 0.82 angstroms and the PS alone averaged a poor 2.7 angstroms, on the same problems from identical initial populations (Table 3). For all three methods, the maximum spatial deviation from the true solution tended to increase as a function of the number of HA (Table 3), but this dependence was much less pronounced with GAPS (Figure 5). Linear regressions of maximum distance against the number of heavy atoms, revealed

Table 2: Mean ( ) and standard deviation (σ) of the correlation (r) between estimated and true Patterson maps averaged over 20 repetitions on each of 11 problem configurations, for a total of 220 runs per #HA.

	Patterson Correlation Coefficient, r					
	GAPS		GA		PS	
#HA		σ		σ		σ
1	0.9999	4.55e-11	0.9989	8.30e-4	0.9918	3.11e-2
2	0.9988	4.53e-3	0.9962	5.67e-3	0.9700	1.68e-2
3	0.9990	4.31e-3	0.9950	4.45e-3	0.9767	1.15e-2
4	0.9996	1.02e-3	0.9967	1.56e-3	0.9802	7.49e-3
5	0.9993	1.13e-3	0.9962	1.72e-3	0.9894	4.75e-3

Table 3: Mean ( ) and standard deviation (σ) of the maximum Euclidean distance between estimated and true atomic locations of the heavy atoms (HA) in the AU averaged over 20 repetitions on each of 11 problem configurations, for a total of 220 runs per #HA.

	Maximum Distance (Angstroms)					
	GAPS		GA		PS	
#HA		σ		σ		σ
1	2.46e-5	2.76e-5	4.64e-2	1.24e-2	2.92e-1	8.92e-1
2	9.92e-2	3.60e-1	2.44e-1	3.04e-1	1.76e+0	4.68e-1
3	9.80e-2	3.72e-1	3.66e-1	2.36e-1	2.34e+0	4.72e-1
4	1.20e-1	2.82e-1	5.20e-1	1.55e-1	2.78e+0	4.28e-1
5	3.10e-1	3.28e-1	8.18e-1	2.14e-1	2.74e+0	3.48e-1

Table 4: Mean number of HA's correctly identified per replication and the percentage (%) of trials that correctly identified all HA's averaged over 20 repetitions on each of 11 problem configurations, for a total of 220 runs per #HA.

	Average # HA's found			% of Trials correctly finding all HA's		
#HA	GAPS	GA	PS	GAPS	GA	PS
1	1.00	1.00	1.00	100.0	100.0	100.0
2	2.00	1.99	1.95	100.0	99.5	95.0
3	2.99	2.94	2.63	98.6	94.1	63.2
4	3.97	3.87	3.20	96.8	87.3	39.1
5	4.81	4.63	3.71	85.5	68.2	28.6

that the slope of the GAPS best fit line (R^2=0.80) was 1/3 that of the GA best fit line (R^2=0.97) and 1/9 that of the PS best fit line (R^2=0.82), as shown in Figure 5. It should be noted that when an estimated HA is more than about 2 angstroms away from the closest true HA in our simulated crystals, the fitness function flattens out. This is probably why the already very poor spatial resolution of the PS does not continue to degrade linearly as the number of heavy atoms increases (Figure 5).

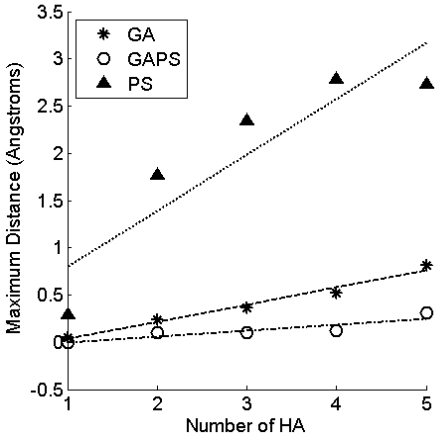

Figure 5: Maximum Euclidean distance between estimated and true atomic locations in the AU, averaged over 20 replications on each of 11 problem configurations, for a total of 220 runs per number of HA. Best fit lines are also indicated.

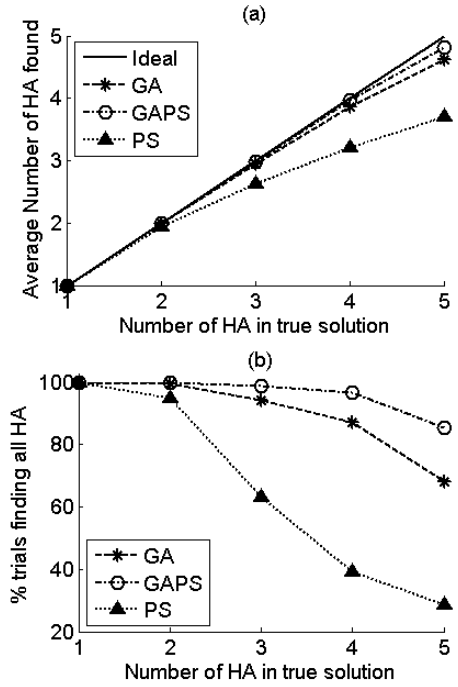

Figure 6: (a) Number of heavy atoms (HA), averaged over all replications and problems with a given number of HA. The solid line represents the ideal, where all heavy atoms are found in every run. (b) Percent (%) of trials that found all heavy atoms, determined over all replications and problems with a given number of HA.

For the simple 1HA problems, all methods were successful in finding the single heavy atom (Table 4). However, as the number of HA increased, GAPS became increasingly more successful than either GA or PS at finding the HA, both in terms of the average number of HA found per trial as well as the percent of trials that were able to find all HA (Table 4, Figure 6). The average number of HA found was significantly higher ($p < 0.005$) for GAPS in comparison to GA, for problems of 3 or more HA, or in comparison to PS, for problems of 2 or more HA, (Table 4, Figure 6a). For the most difficult problems tested (5HA), GAPS found 4.8 HA on average, finding all 5 HA over 85% of the time (Table 4). Success rate in finding HA appears to decrease non-linearly in all the methods. For problems with more than 2HA, the frequency with which the PS algorithm was able to find all the HA dropped dramatically (Figure 6b). However, the rate of this performance drop is lower in GAPS than in GA (Table 4, Figure 6b), indicating that the second stage PS in the hybrid algorithm was not only helping to improve spatial resolution, but was also helping to find additional HA.

Factors affecting runtime are shown in Table 5. The average number of generations required for the GA to converge (either alone or as stage 1 of GAPS) increased linearly (R^2=0.99) with the number of HA in the problem, however the number of function evaluations increased quadratically for both GA and GAPS, reflecting our quadratic population sizing model. The number of function evaluations for the PS also increased quadratically, whether applied to the best individual from the GA (i.e., in stage 2 of the GAPS) or whether applied to the best of the initial random population. However, PS required 1-2 orders of magnitude fewer function evaluations than did the GA (since it was only optimizing a single individual). For reference, GAPS required about 3.5 hours to solve a 5HA problem on a 2.2 GHz Pentium IV.

Table 5: Mean number of generations for GA convergence and mean number of function evaluations per replication, averaged over 20 repetitions on each of 11 problem configurations, for a total of 220 runs per number of HA.

# HA	# gens for GA	# of Function Evaluations			
		GA	GAPS	PS - stage 2 of GAPS	PS-alone
1	124	12,425	12,791	366	432
2	216	64,825	66,877	2,052	1,853
3	309	185,345	189,892	4,547	4,304
4	457	457,022	465,497	8,475	8,703
5	554	830,495	846,311	15,816	18,161

4. DISCUSSION AND CONCLUSIONS

The heavy atom location problem, used in crystallographic phasing for the determination of the molecular structure of proteins, is a multi-modal inverse problem. The non-linearity of

the problem increases rapidly with the number of heavy atoms. In the P422 space group chosen here, there are eight global optima ("niches"), and crosses between individuals from different niches will yield low fitness solutions. Unfortunately, explicit niching approaches, such as fitness sharing [5] and crowding [11], are not practical in this application domain, because of the difficulty in determining an appropriate similarity metric. This is because there is no practical way to determine the hand and origin of a potential solution, at least not until the solution nears convergence, and it is meaningless to measure Euclidean distances between atoms from different hands and origins. We considered using an island model, but preliminary experimentation with subpopulations connected in a ring topology did not prove effective for this problem [4].

In addition, the problem is only very weakly decomposable into locating individual atoms, since significant epistatic interactions exist between the atoms in the solution. This epistasis arises due to the cross peaks in the Patterson map. To better understand this, consider the following. Each heavy atom located in the AU, from one of the eight possible hands or origins, maps to eight heavy atoms in the unit cell, and therefore maps to 28 (8 choose 2) self-peaks in the Patterson map. However, suppose we are searching for two heavy atoms, and we have correctly located one heavy atom in one hand and origin and a second heavy atom in a different hand and origin. Although there will be 56 (28+28) correct self-peaks in the Patterson map, there could be up to 64 (8×8) *in*correct cross-peaks, therefore resulting in poor correlation coefficient with the (potentially overlapping) peaks in the true Patterson map. The number of self peaks increases linearly with the number of heavy atoms in the AU, but the number of cross peaks increases with the number of heavy atoms in the AU choose 2. Thus, the epistatic component of the fitness function increasingly dominates the linear component as the number of heavy atoms in the AU of the true solution increases.

The goal of this pilot study, which grew out of a class project [4], was to explore the potential for using a hybrid GA to search this difficult landscape. Our results show that a 2-stage hybrid algorithm, using a stochastic genetic algorithm for stage 1 followed by a deterministic pattern search algorithm for stage 2, can successfully locate up to 5 heavy atoms in small computer simulated crystals using noise free data. The stage-1 GA identifies most of the heavy atoms. Subsequent refinement of the best solution using a PS improves the spatial resolution of the estimate and also, in most cases, finds any remaining heavy atoms not found by the GA. In contrast, PS alone, when applied to the best of a random population of solutions, is not capable of reliably finding more than one heavy atom.

Several simplifications were introduced into this study. For one thing, we assumed that the number of heavy atoms was known in advance. Relaxing this assumption could be handled in a number of ways, including: a) simultaneously estimating a subset of the heavy atoms, determining their hand and origin, and then refining the estimate in the given hand and origin by adding in heavy atoms one at a time until an optimum number is found, b) using variable length chromosomes, or c) running different repetitions with different assumed numbers of heavy atoms and picking the best.

Another simplification we employed was to use noise free data for amplitudes of structure factors due to heavy atoms alone. In reality, such data can be quite noisy. One source of this noise is simply measurement error. Another source of noise comes from the fact that these amplitudes cannot be directly measured, but are instead estimated by subtracting measured amplitudes due to protein crystals alone from measured amplitudes of protein crystals doped with heavy atoms, and the presence of the heavy atoms will slightly perturb the locations of the protein atoms. While noise free data permitted us to use a conceptually simple fitness function based on Patterson map correlation, other functions would likely be more effective with real data. For example, correlation functions in *reciprocal space* (the transform of the Patterson map) have been used successfully [6],[18],[19] and are more generally applicable to lower symmetry space groups. A maximum likelihood approach has been shown to be effective in helping to minimize the effects of noise [19], and this could be incorporated into a more sophisticated fitness function.

We have also simplified the problem by limiting the number of heavy atoms, the size of the crystal, the range of locations for the true heavy atoms, and the number of reflections. Relaxing these simplifications should be relatively straightforward, but will add significantly to the computational burden of the estimation process by increasing the size of the search space and slowing down computation of the fitness function (the runtime of which is a function of the number of reflections). Appropriate handling of solutions near boundaries or symmetry elements [19] would also be necessary to make this program more realistic, but were not implemented here for the sake of computational simplicity. Generalizing the approach for other space groups is also conceptually straightforward, although the code will be complicated by certain details such as variable bounds on the AUs in some space groups. Additionally, the number of global optima will vary for different space groups, with possible implications on required population sizes.

One way to potentially significantly *reduce* the computational requirements of the search would be to seed the initial population with likely candidate locations for heavy atoms, as in [6]. Likely candidates could be determined by doing a one-time evaluation of a set of finite possible atoms locations (e.g., discrete points on a regular grid imposed on the AU), by searching for the self-peaks of these potential locations in the Patterson map. A recombination-based GA could then be used to find promising subsets of these potential locations, with subsequent PS to refine the spatial resolution of the estimated atoms.

In summary, our results indicate that a hybrid 2-stage genetic algorithm with subsequent pattern search may be a viable approach for locating heavy atoms in a consistent hand and origin, for crystallographic phasing using iso-morphous replacement. Further research along these lines is warranted.

5. ACKNOWLEDGMENTS

This work was supported in part by a pilot award funded by DOE-FG02-00ER45828 awarded by the US Department of Energy through its EPSCoR Program and an instructional incentive grant from the University of Vermont Center for Teaching and Learning. We thank Joshua Gilbert, Jim Hoffmann, and all the students in the fall 2004 course in evolutionary computation at the University of Vermont [4], whose helpful ideas and discussions contributed to our approach. We would especially like to acknowledge M.A. Rould, for lending his expertise in crystallography.

6. REFERENCES

[1] Chang, G. & Lewis, M. Using genetic algorithms for solving heavy-atom sites. *Acta Crystallographica Section D* 50, 667-674 (1994).

[2] Chang, G. & Lewis, M. Molecular replacement using genetic algorithms. *Acta Crystallographica Section D* 53, 279-289 (1997).

[3] de Graaff, R.A., Hilge, M., van der Plas, J.L., and Abrahams, J.P. Matrix methods for solving protein substructures of chlorine and sulfur from anomalous data. *Acta Crystallographica Section D* 57, 1857-1862 (2001).

[4] Eppstein, M.J. and Hoffmann, J.P. Crystallographic Case Study in an Interdisciplinary Evolutionary Computation Course, accepted to ECP track, *Genetic and Evolutionary Computation Conference* (2005).

[5] Goldberg, D.E., Richardson, J. Genetic Algorithms with Sharing for Multimodal Function Optimization. In *Proc. 2nd International Conference on Genetic Algorithms and their Applications.* pp. 41-49. New Jersey (1987).

[6] Grosse-Kunstleve, R.W. & Adams, P.D. Substructure search procedures for macromolecular structures. *Acta Crystallographica Section D* 59, 1966-1973 (2003).

[7] Hart, W.E. Evolutionary pattern search algorithms for unconstrained and linearly constrained optimization. *IEEE Transactions on Evolutionary Computation*, vol. 5 No. 4. (2001).

[8] Hart, W.E. and Hunter, K.O. A performance analysis of evolutionary pattern search with generalized mutation steps. *Proc. Congress on Evolutionary Computation*. pp. 672-679. (1999).

[9] Kissinger, C.R., Gehlhaar, D.K. & Fogel, D.B. Rapid automated molecular replacement by evolutionary search. *Acta Crystallographica Section D* 55, 484-491. (1999).

[10] Lewis, R.M. and Torczon, V. Pattern search algorithms for bound constrained minimization, *SIAM Journal on Optimization.*, 9(4): 1082-1099, (1999).

[11] Mahfoud, S.W. Crowding and preselection revisited. *Proc. Of the 2nd Conference on Parallel Problem Solving from Nature.* pp. 27-36. Amsterdam (1992).

[12] MathWorks. 24 Prime Park Way, Natick MA 01760-1500 (2004).

[13] Miller, S.T., Hogle, J.M. and Filman, D.J. A genetic algorithm for the ab initio phasing of icosahedral viruses. *Acta Crystallographica Section D* 52, 235-251 (1996).

[14] Moscato, P.A. On evolution, search, optimization, genetic algorithms, and martial arts: towards memetic algorithms. *Technical Report Caltech Concurrent Computation Program Report 826*. Caltech, Pasadena, California (1989).

[15] Regué, J.R., Ribó M., & Garrell, J.M. Radiated emissions conversions from anechoic environment to OATS using a hybrid genetic algorithm – gradient method. *2001 IEEE Symposium on Electromagnetic Compatibility Record*. pp. 325-329. Montreal (2001).

[16] Rhodes, G. *Crystallography Made Crystal Clear*. Academic Press: San Diego (2000).

[17] Sands, D.E. *Introduction to Crystallography*. Dover Publications, Inc: New York (1975).

[18] Schneider, T. R. and Sheldrick, G. M. Substructure solution with SHELXD. *Acta Crystallographica Section D* 58, 1772-1779 (2002).

[19] Terwilliger, T. C. and Berendzen, J. Automated MAD and MIR structure solution *Acta Crystallographica Section D* 55, 849-861. (1999).

[20] Trabia, M.B. A hybrid fuzzy simplex genetic algorithm. *Proc. ASME Design Engineering Technical Conferences* (2000).

[21] Wetter, Michael, and Wright, Jonathan. Comparison of a generalized pattern search and a genetic algorithm optimization method. *Proc. 8th International Building Performance Simulation Association Conference* vol III, pp 1401-1408. (2003).

[22] Van Laarhoven, P.J.M. and Aarts, E.H.L. *Simulated Annealing: Theory and Applications*. Kluwer Academic Publishers: Norwell, MA. (1987).

[23] Vazquez, M., and Whitley, D.L. A Hybrid Genetic Algorithm for the Quadratic Assignment Problem. *Genetic and Evolutionary Computation Conference.* pp. 169-178 (2000).