Generalized Schemata Theorem Incorporating Twin Removal for Protein Structure Prediction

Md Tamjidul Hoque, Madhu Chetty, and Laurence S. Dooley

Gippsland School of Information Technology (GSIT)
Monash University, Churchill VIC 3842, Australia
{Tamjidul.Hoque,Madhu.Chetty}@infotech.monash.edu.au,
lsdaussie@ieee.org

Abstract. The schemata theorem, on which the working of Genetic Algorithm (GA) is based in its current form, has a fallacious selection procedure and incomplete crossover operation. In this paper, generalization of the schemata theorem has been provided by correcting and removing these limitations. The analysis shows that similarity growth within GA population is inherent due to its stochastic nature. While the stochastic property helps in GA's convergence. The similarity growth is responsible for stalling and becomes more prevalent for hard optimization problem like *protein structure prediction* (PSP). While it is very essential that GA should explore the vast and complicated search landscape, in reality, it is often stuck in local minima. This paper shows that, removal of members of population having certain percentage of similarity would keep GA perform better, balancing and maintaining convergence property intact as well as avoids stalling.

Keywords: Schemata theorem, twin removal, protein structure prediction, similarity in population, hard optimization problem.

1 Introduction

Protein structure prediction (PSP) using lattice model is regarded as a very hard optimization problem. This is because the prediction using lattice model is proven to be NP-complete [1],[2] and the number of possible valid (i.e., self avoiding walk) conformation is astronomical [3], [4]. We have chosen Genetic Algorithm (GA) as a vehicle for providing solution to the *protein structure prediction* (PSP) problem for its performance in various domains [5], [6], [7], [8], [9], [10], [11], [12], [13], [14]. Crossover, regarded as the key operation of GA, is also being adapted by almost all other promising search approaches [15],[16],[5],[17],[12]. It is considered as the potential operation that can build a promising conformation by cutting and joining the potential sub-parts of more than one conformation. In some cases, the GA population strategy is also being adapted by other approaches. While GA performance is generally very effective it can sometimes stall [18] in a hard optimization problem [19] like PSP with the protein sequences having length above, say 30 [12]. Thus, like other promising approaches, GA too cannot ensure the final generation to contain an

J.C. Rajapakse, B. Schmidt, and G. Volkert (Eds.): PRIB 2007, LNBI 4774, pp. 84–97, 2007.

optimal solution. Even, effective [18],[20] elitism can become ineffectual for PSP problem. This problem is so difficult that unlike other type of problems, a mere application of any of the known approaches will not provide improved results. Therefore, in this paper, we present the generalization of the schemata theorem by incorporating twin removal which is necessary to overcome the limitations of the GA and show the impact of this generalization upon GA operation in order to secure more accurate and efficient PSP solutions. To achieve this, in the initial stage we revisit the idea of identical chromosomes (twins) in the population and relax the concept to embrace similar (strongly-correlated) chromosomes. This helps to generalize the schemata theorem as well as to find the percentage of similarity within the population that can keep in GA optimum search condition.

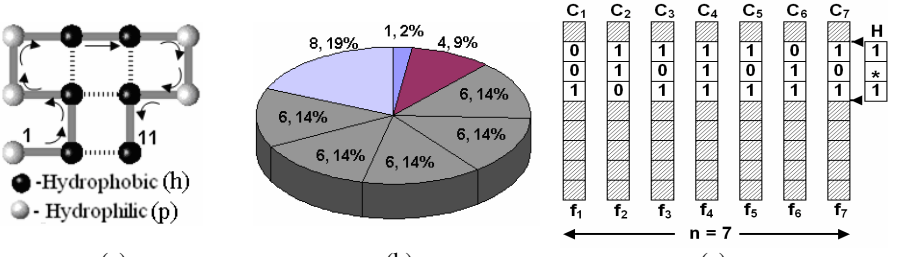
2 Twins in GA Population

The *schemata theorem* as the basis of a GA, has had its critics as evidenced in [21], [22]. The mathematical derivations in relation to the schemata theorem supports that the schemata with above average fitness values would most likely be sustained as the generations proceed and consequentially the similarity [23], [24], [25],[26],[27] grows within the population. This means that although we can set the crossover rate to a desired value, in many cases, the operation generates no variation due to the similarity. Earlier, it was observed [28] that due to the 'stochastic error' associated with GA's genetic operators, the genetic algorithm tends to converge to a single solution. This can raise two different issues. First, there are certain applications where search interest is not for one but several solutions [29], such as to find Pareto front on a problem using multi-objective optimization application. Second, convergence to a single solution means the search becomes stagnant which can be due to the population losing its diversity. This phenomena is termed as 'genetic drift' [29], [30] due to which, in hard optimization problem such as PSP, the search space is extremely convoluted. It can cause the aforementioned stall effect which could be devastating. The searches can get stuck in local minima without exploring much of the vast space.

The existence of twins and the requirement for their removal in a GA is not new. This matter appears as diversity issue in literature as a result of growth of the twins. The growth of twins was considered [26] in evaluating the cost of duplicate or identical chromosomes which suggested starting each chromosome with different patterns to avoid twins. However, if twin growth is inherent in a GA search, then the effect of initialization using different patterns will quickly decline after starting. In [23], [31], it was advocated that if a population comprised all unique members, tests need to be continually applied to ensure that identical chromosomes did not breed. If chromosome similarity does not grow, then the GA may not converge because the search process becomes random rather than stochastic. While if it does grow, and then finding a non-similar chromosome to mate with, clearly becomes rare because of the inevitable occurrence of more twins, and the increasingly costly exercise of finding dissimilar chromosomes. On the other hand, it was also advocated [28] to allow individuals to reproduce if they are very closely similar. But, we have shown [32] that crossover between identical chromosome is a mutation operation which can turn a

stochastic search approach indirectly into a random search, specially for complex problem and therefore the solution of the problem rarely converges [11],[12].

Aforementioned issues related to twin removal provide motivation for the investigations presented in this paper. The need for twin removal was originally highlighted in [25] which emphasized that duplicates chromosomes (*twins*) reduce diversity and ultimately lead to poorer performance. The study was confined solely however, to the detection and removal of identical chromosomes that were unique to each other, with no consideration being given to the removal and impact of similar chromosome or strongly correlated chromosomes. To mitigate the limitations caused by the stall condition, PSP using a GA has principally been confined to developing models based around special operators [33],[34] statistical approaches [5],[33],[35] and special treatment techniques such as *cooling* [11],[12] constraints and hybridization [10],[14],[15],[36] with the consequence that resulting GA-based solutions are both model and sequence dependent but are never generic. Therefore, generic improvement can be coupled for further improvement.



(a) (b) (c)

Fig. 1. (a) Conformation of sequence *phhpphhpphh* in 2D HP model [37] is shown by solid line. Any two hydrophobic residues being *topological neighbor* (TN) is indicated by dotted line. Fitness = -(TN Count) = -4, here. Three different arrows indicating *Left* (0), *Right* (1) and *Forward* (2) can be used to for chromosome encoding and it forms 001122110 in this case. **(b)** Pie chart of population having fitness 8, 6, 6, 6, 6, 6, 4 and 1. *Legend*: Fitness, Fitness % (with respect to the sum of the fitness values) **(c)** An example schema, H [1∗1] at bits 2 to 4, contained in chromosomes 3, 4, 5 and 7 of population size $Pop_z = 7$ at generation t.

A *chromosome correlation factor* (CCF) defines the degree of similarity existing between chromosomes. For similarity measurement between two individuals in the genotype as described in [29], we also measure it by counting the number of bits along each chromosome that are equal in the two individuals being compared. For chromosome presented [32] in the 2D HP (used in this paper) for PSP problem, three bit code 0, 1 and 2 are used for presenting three moves (see Fig. 1 (a) description).

It will be shown that by removing chromosomes having a similarity value greater than or equal to CCF during the search process enables the GA to continue seeking potential PSP solutions and ultimately provide superior results. The improved PSP performance of the algorithm based upon the generalized schemata theorem is analyzed upon accepted benchmark PSP sequences [34],[38]. Randomly-selected single point crossover and mutation operations are used in this paper as well as in the literature [11],[12] for PSP. This is because, as the solution becomes phenotypically

compact it can produce more collisions [14],[16], if multi-point crossovers and mutations were involved which would leading to increasing collisions that produce non-self-avoiding walks within the conformation.

3 Preliminaries of Schemata Theorem

While this paper considers only the *Simple GA* (SGA), without any loss of generality, the theoretical framework developed is applicable to all GA variants [39]. Firstly, the initial population is generated, where the i^{th} chromosome C_i is selected based on the fitness f_i with probability $(f_i / \bar{f})$, where $\bar{f}$ is the average fitness of the population. Parents then produce offspring by crossover at a rate p_c for a population of size Pop_z, with the generated offspring chosen with a *selection* probability $(f_i / \bar{f})$ and a mutation rate p_m. Usually, a small percentage of *elite* (high fitness) chromosomes are then copied to the next generation to retain potential solutions, with any remaining chromosomes unaffected by crossover, mutation or elitism moved to the next generation. Assume, an alphabet of cardinality |*A*| (defined as b_{count} in this paper) is used and hence the cardinality of schema is (|*A*|+1) including the *don't-care* which is normally applied to cover the unrestricted locus of the schema. The length of the schema $\delta(H)$ is the distance between the position of the first and last non *don't-care* characters, which actually indicates the number of possible crossover positions. For a chromosome length n, there are $((|A|+1)^n - 1)$ possible schema, excluding the combination comprising only *don't cares*, so a population of Pop_z chromosomes evaluates up to $((Pop_z)((|A|+1)^n - 1))$ schema, which provides implicit parallelism within the GA search. The order of schema $o(H)$ is the number of non *don't-care* characters, which governs the impact that any mutation has upon the schema. The number of occurrences of schema H in a population Pop_z at time t (which equals the number of generations) is given by $m(H,t)$. Throughout this paper, *twins* refer to pairs of chromosomes which are, with respect to their conformations, either *i*) identical, so CCF = 1, or *ii*) correlated with CCF ≥ *r,* where *r* is the minimum admissible level of similarity defined for a population. Also, the term *overall similarity* is used to indicate the average of all CCF values of any chromosome with respect to all the other chromosomes in the population.

4 Limitations of the Schemata Theorem

In the following sections, limitations of the working principles of GA, i.e., *schemata theorem* [40], has been explored in the context of twin removal.

Selection: For a chromosome C_k having fitness f_k, the probability of C_k being selected by roulette wheel selection, is given by (1):

$$p_k = f_k \Big/ \sum_{i=1}^{Pop_z} f_i \tag{1}$$

The proportionate *selection* probability of the first chromosome (see Fig. 1 (b)) will be $p_1 = (8/43)$, and similarly $p_2 = (6/43)$, …, $p_8 = (1/43)$. This is fallacious however, as from the pie-chart in Fig. 1 (b), it is clear that assuming chromosomes having the same fitness are identical, the fitness 6 occupies 68% in total, so the probability of a rolling marble randomly selecting a segment having fitness 6 is expressed as $p_{effective_2} = \sum_{i=2}^{6} p_i = 30/43$. The *effective selection* probabilities for C_1 (C_2 or C_3 or C_4 or C_5 or C_6), C_7 and C_8 are thus $8/43$, $30/43$, $4/43$ and $1/43$ respectively. Effectively, any of the fitness 6 occupies 70% of the pie-chart instead of 14%. Now consider an arbitrary schema H [1∗1] from bit position 2 to 4 as shown in Fig. 1 (c). The number of occurrences of such schema at time t is, $m(H, t) = 4$. The expected number of occurrences at time $(t+1)$ is $m(H, t+1)$ which depends on the fitness of the chromosomes containing the schema H such as C_3, C_4, C_5 and C_7. Hence, $\bar{f}(H,t)$ = (f_3 + $f_4 + f_5 + f_7$) / 4. The average fitness $\bar{f}$ is now defined as:

$$\bar{f} = \left(\sum_{i=1}^{Pop_z} f_i \Big/ Pop_z \right) \tag{2}$$

So if $\bar{f}(H,t) > \bar{f}$, then the number of occurrences of schema H in the next generation is likely to increase by $(\bar{f}(H,t)/\bar{f})$. Thus, the expected number of occurrences of schema H at time $(t+1)$ can be expressed as:

$$m(H, t+1) = m(H, t) \frac{\bar{f}(H,t)}{\bar{f}} \tag{3}$$

where, $\bar{f}(H,t)$ is the average fitness of chromosomes containing schema H.

Crossover: The *schemata theorem* computes the probable occurrences of a particular schema H in the next generation, with the proviso that the longer the schema length, the greater probability that the H will be disrupted by either a crossover or mutation operation. For a chromosome of length n there are $(n-1)$ possible crossover positions. Therefore the *disruption* probability is $(\delta(H)/(n-1))$ with the complementary *existence* probability being $(1-(\delta(H)/(n-1))$, so in general the lower bound of the *existence* probability p_e having a crossover probability p_c is:

$$p_e \geq \left(1 - p_c \frac{\delta(H)}{n-1} \right) \tag{4}$$

The derivation of (4) comes from the fact that if a crossover point lies within the region of schema H, then the schema does not remain intact in the offspring, though this is not always the case. Section **5**, examines all the various possible scenarios:

Mutation: The mutation operation is able to disrupt any schema. With a *mutation* probability p_m, the bit *disruption* probability of a bit or character changing is $(1-p_m)$, so for the schema H having order $o(H)$, the *existence* probability of H is:

$$p_e = (1 - p_m)^{o(H)} \tag{5}$$

For very small values of p_m ,

$$p_e \approx (1 - p_m\ o(H)) \tag{6}$$

Schemata Theorem: The number of occurrences of schema H in (3) can be expressed using (4) and (6) as:

$$m(H,\ t+1) = m(H,\ t)\frac{\bar{f}(H,t)}{\bar{f}}\left(1 - p_c\frac{\delta(H)}{n-1} - p_m\ o(H)\right) \tag{7}$$

which was the formal mathematical representation of the *schemata theorem*. But, as (4) is incomplete then so also is (7). While it is readily apparent that (7) supports the growth of similarity within a population, it fails to reflect certain anomalies within the original *schemata theorem* that can impact significantly upon GA operations, as the growth in twins and their potential deleterious effect in complex landscape applications such as PSP are considered.

5 Generalization of the Schemata Theorem

To analyse the effect of growing similarity in a population, the following sections directly address the particular limitations highlighted in this Section by firstly generalizing the selection process, resolving the issue of the crossover component

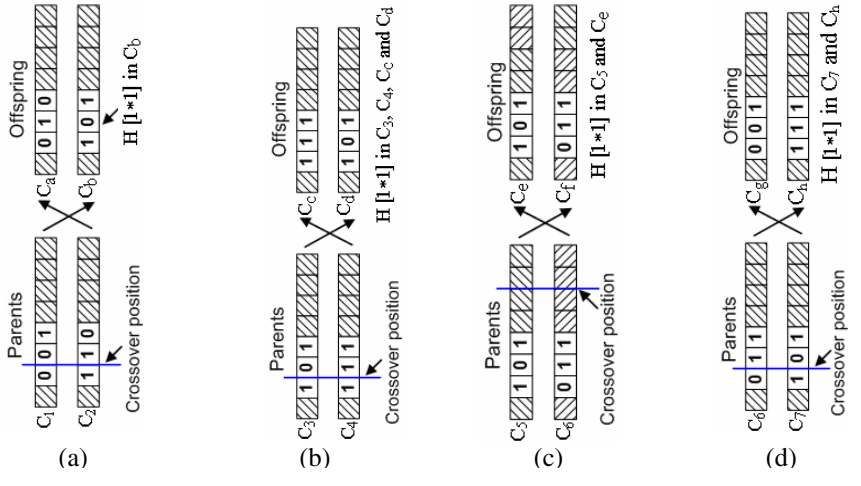


Fig. 2. (a) Schema H [1 * 1] produced in offspring even when both parents do not have that schema. **(b)** The offspring always contain schema H [1 * 1] irrespective of the crossover position when both the parents have the particular schema. **(c)** One of the parents contains schema H [1 * 1] and the crossover positions lies outside the schema region. **(d)** One of the parents contains schema H [1 * 1] and the crossover positions is inside the schema region.

contributing in equation (7) and then integrating the solutions into a *generalized schemata theorem* framework.

Selection: The scenario under consideration is that the number of highly fitted chromosome will become larger as they are increasingly selected for crossover and mutation in each generation. The selection procedure will always favor those similar chromosomes that are higher in number in the population, so if w_k is the number of such similar chromosomes having fitness f_k, it will have a lower bound of unity and compared with (1), the effective selection now becomes:

$$p_k = \left(w_k f_k \Big/ \sum_{i=1}^{Pop_z} f_i \right) \tag{8}$$

which is a generalized representation of the original [40] selection procedure with $w_k = 1$. This is fully supported by the comparative examples in **Fig. 1** (b), where the selection process anomaly highlighted in Section **4**, mandates an appropriate twin removal strategy be implemented in order to ensure that as w_k tend to 1, the core *schemata* theory is upheld.

Crossover: The crossover operation however, may in certain cases not be disruptive [24], which can be interpreted as providing an *Accrued Benefit* (*AB*) because the schema of interest *H* is preserved rather than disrupted, which is not reflected by (4). Three *AB* scenarios are identified:

i) Accrued Benefit$_1$: Neither Parent Contains a Particular Schema
Consider the scenario illustrated in Fig. 2(a) of the crossover between two parents that do not contain schema H, though H may be expected to be created in the offspring. As neither of the parents contain schema H the crossover must occur within the schema region to create such an offspring, so the resulting *AB* can be expressed as in (9).

$$AB_1 = \left[1 - \left(\sum f(H,t) \Big/ \sum_{i=1}^{Pop_z} f_i \right) \right]^2 \left(\frac{\delta(H)}{n-1} \right) \Delta \tag{9}$$

The square parenthesis term is the *selection* probability of those parents that do not contain schema H, with $\sum f(H,t)$ being the aggregated fitness values of those chromosomes containing *H*, so the *selection* probability using (8) of these particular chromosomes is $\left(\sum f(H,t) \Big/ \sum_{i=1}^{Pop_z} f_i \right)$. The second term in parenthesis represents the probability of the crossover point existing within region H, where it is intuitively reasonable to assume both parents contain some part of schema H close to the crossover point, and this is given by probability Δ. To estimate Δ, assume a single crossover point divides schema H into H_1 and H_2, that is the schema is actually a concatenation of sub-schema so,

$$H = H_1 \bullet H_2 \text{ and } \Delta = \left(b_{count}\right)^{-(o(H_1)+o(H_2))} = \left(b_{count}\right)^{-(o(H))} \tag{10}$$

since $o(H) = o(H_1) + o(H_2)$. In the example in Fig. 2 (a) where the crossover occurs between positions 2 and 3, the schema $H[1*1]$ is divided into $H_1[1]$ and $H_2[*1]$, where $o(H_1) = 1$ and $o(H_2) = 1$. As $A = \{0,1\}$ then, $b_{count} = 2$ and $\Delta = 2^{-1}.2^{-1} = 0.25$. An important point in (10) is, for a fixed $\delta(H)$, Δ directly depends upon chromosomal encoding and proportional to b_{count}.

ii) Accrued Benefit$_2$: Both the Parents Contain a Particular Schema
If both the parents contain schema *H* as shown in Fig. 2 (b), then *H* will never be lost by crossover irrespective of the crossover position. So,

$$AB_2 = \left(\sum f(H,t) \Big/ \sum_{i=1}^{Pop_z} f_i \right)^2 \tag{11}$$

and since the generation continues, this *benefit* increases due to increments in the similarity, which will assist in the growth of twins.

iii) Accrued Benefit$_3$: Only One Parent Contains the Schema
In this case, two options are feasible when one parent contains schema H and the other does not.

(a) Crossover Point is Located Outside the Schema Region
Since the crossover point does not lie within the schema region (**Fig. 2** (c)), then:

$$AB_{3a} = \left(\left(\sum f(H,t) \right) \Big/ \left(\sum_{i=1}^{Pop_z} f_i \right) \right) \left\{ 1 - \left(\left(\sum f(H,t) \right) \Big/ \left(\sum_{i=1}^{Pop_z} f_i \right) \right) \right\} \left(1 - \frac{\delta(H)}{n-1} \right) \tag{12}$$

where the third term in parenthesis indicates the probability that the crossover point is not located within the schema length and region.

(b) Crossover Point Lies Within the Schema Region
The crossover point now lies within the schema region (Fig. 2(d)) and it is further assumed that the crossover point divides the schema *H* into H_1 and H_2 for single crossover position, so $H = H_1 \bullet H_2$ and:

$$AB_{3b} = \left(\left(\sum f(H,t) \right) \Big/ \left(\sum_{i=1}^{Pop_z} f_i \right) \right) \left\{ 1 - \left(\left(\sum f(H,t) \right) \Big/ \left(\sum_{i=1}^{Pop_z} f_i \right) \right) \right\} \theta \tag{13}$$

$$\text{where, } \theta = \left(\left(\delta(H) \right) / (n-1) \right) \left\{ (b_{count})^{-o(H_1)} \oplus (b_{count})^{-o(H_2)} \right\} \tag{14}$$

where $\oplus$ is the 'Exclusive OR' operation, while θ represents the probability of the formation of schema *H* from parents. The first bracketed term in (14) is actually the probability of the crossover point occurring within the schema region, while the second term is the probability that part of schema *H* will come from each parent, so *H* resides exclusively in one of the offspring. The composite AB_3 for the case where only one parent contains the schema now becomes:

$$AB_3 = AB_{3a} + AB_{3b} \tag{15}$$

Combining the three *Accrued Benefit* from (9), (11) and (15), the *existence* probability p_e of a schema due to crossover occurring at a rate p_c can be expressed as:

$$p_e = p_c\left(AB_1 + AB_2 + AB_3\right) \tag{16}$$

The Generalized Schemata Theorem: The equations delineated in the previous two sections covering chromosome selection and crossover, are now formally embedded into a *generalized schemata theorem* framework. With a *crossover* probability $p_c < 1.0$, those chromosomes unaffected by crossover occur at $(1 - p_c)$. So the original *schemata theorem* in (7) can be rewritten using (10) as in (17):

$$\begin{aligned} m(H,t+1) = \left[(1-p_c)\, m(H,t)\, \frac{\bar{f}(H,t)}{\bar{f}}\right] + m(H,t)\, \frac{\bar{f}(H,t)}{\bar{f}}\, p_c\left(AB_1 + AB_2 + AB_3\right) \\ - m(H,t)\, \frac{\bar{f}(H,t)}{\bar{f}}\, p_m\, o(H) \end{aligned} \tag{17}$$

Now, (17) is a generalized representation of how the GA functions, such that in the case where p_c=1.0, AB_1= 0, AB_2= 0, AB_{3b} = 0 and all chromosome selection probabilities are ignored (the first two terms in (17) then it reduces to the classical schemata theorem. Interestingly (17) supports the nonlinear fast growth of the surviving (also referred to as favorable) schema and with the incorporation of the appropriate selection procedure and various crossover scenarios, (17) clearly reveals the obvious expansion of twins in the population. The implications of this growth and the increasing likelihood of converging prematurely into the stall condition are now considered.

Impact of Generalization: The inexorable growth of identical and also progressively more highly-correlated twins as manifest in (17) can lead to the premature convergence or stall [18],[41] in the search process, a situation exacerbated by crossover creating even more twins and the impact of the mutation becoming increasingly ineffectual. These two issues are now respectively considered in the context of the new generalized framework.

(a) <u>*Premature Convergence or Stall Condition*</u>: The *reproduction* probability of twins ($r \leq$ CCF ≤ 1) can be expressed using (8) as:

$$P(C_k, C_k) = \left(p_k\right)^2 \tag{18}$$

So, the number of twins that are going to be in the next generation can be written as:

$$Count(C_k, t+1) = P(C_k, C_k)\left(Pop_z\right) p_c \tag{19}$$

Now consider the case where the number of similar chromosomes becomes close to the population so $w_k \approx Pop_z$ and $w_k f_k \approx \sum_{i=1}^{Pop_z} f_i$ in (8), so using (18) we get:

$$P(C_k, C_k) \approx 1 \tag{20}$$

which is the stall or premature convergence condition. (19) shows that nearly all offspring generated throughout the population will be similar and go forward to the

next generation with the result that there will be no variation in subsequent generations. It is entirely reasonable therefore to surmise that as the *effective* crossover rate $p_c \approx 0$, strategies that facilitate efficient removal of both identical and highly correlated twins will improve the GA performance, a premise that is fully corroborated in the experimental results Section 6.

(b) *Ineffective Mutation*: The growth of correlated twins inevitably weakens the impact of mutation, which despite introducing random variations and thereby different schemas, will quickly disappear in the midst of the common schema of so many correlated chromosomes in the population such that when $w_k \rightarrow Pop_z$, the chromosome selected for mutation ($C_{mutated}$) is very likely to be similar (high CCF value) to the majority of the population. By considering the mutation position, if the conformational change differs with respect to C_k, then two principal scenarios arise: *i*) After mutation, $C_{mutated}$ has a lower fitness ($f_{mutated}$) than average, so it is less likely to be selected, and thus will not be in the next generation. *ii*) After mutation, $C_{mutated}$ has a higher fitness than average, but is not similar to highly populated chromosomes, and so while $f_{mutated} > f_k$, as $w_k \rightarrow Pop_z$ the effect due to (8) becomes $f_{mutated} << w_k\ f_k$, so the chances of $C_{mutated}$ being selected for reproduction in the next generation are lower and it is likely the fitter $C_{mutated}$ will die away, so leading to an effective mutation rate of $p_m \approx 0$. While one possible approach to overcoming these issues is to use *elitism* [23], [42] to preserve a small proportion (5% to 10%) of elite chromosomes through the generations, this can convert the GA into a random rather than stochastic search process, with convergence never guaranteed. A better strategy is to remove both identical and highly correlated chromosomes to not only improve the performance of the GA but also avoid premature convergence.

6 Simulation and Experimental Results

Simulations were undertaken with (*TR-r*) and without (*WT*) the twin removal strategy implemented in the population. For twin removal (*TR-r*), it is performed after the crossover and mutation operations, for a range of CCF settings from r = 1.0 (identical chromosomes only) to r = 0.2 (the widest chromosome similarity $0.2 \leq \text{CCF} \leq 1.0$) in steps of 0.1 (e.g., *TR-60* refers to the removal of all chromosome twins having an admissible similarity value of 0.6 (60%) or above). A knock out system was adopted based on the superior fitness value in a *Correlated Twin Removal* (CTR) algorithm (see Algorithm I), where the chromosome with the lower fitness was removed. CTR uses the minimum admissible correlation value r when comparing chromosome pairs for conformational similarity (Line 4), and if twins are identified, the one with the lower fitness is removed (Lines 5 to7). After the removal, the gap is filled by randomly generated chromosomes, which for simplicity are not crosschecked for further twins. The GA parameters [26], [44] for experiments were set as Pop_z = 200, p_c = 0.8, p_m = 0.05 with elite rate = 0.05. *WT* (without twin removal) runs where same as in [12] but without cooling and *TR-100* is the same approach as in [25]. PSP with complex landscape takes longer time to converge. For this reason a maximum of

Algorithm-I: *Correlated Twin Removal* (CTR)
Input: Population size= Pop_z , Chromosome (C) length = n , Minimum admissible correlation = r, where, $CCF \geq r$
Output: Population without twins of size $\leq Pop_z$
Assumption: *RetSimilarity* (i, j) returns % of similarity between C(i) and C(j), where $i \neq j$.

```
1 FOR i = 1 to (Pop_z − 1) DO
2  { IF C(i).MarkDeleted = False THEN
3    FOR j = i+1 to Pop_z DO
4     { IF RetSimilarity(i, j) ≥ r% THEN
5      { IF |C(i).Fitness| < |C(j).Fitness| THEN
6          {Swap (C(i), C(j)) }
7          C(j).MarkDeleted = True } }}
8        }}
```

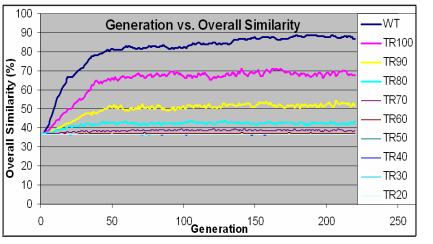


Fig. 1. Generation vs. Overall Similarity (%) plot for PSP of length 50

Table 1. Run results of 5 iterations of PSP for HP sequence length 50; maximum generation = 6000 and minimum fitness = -21. Sequence: H2(PH)3PH4PH(P3H)2P4H(P3H)2PH4P(HP)3H2, [38].

WT	*TR-100*	*TR-90*	*TR-80*	*TR-70*	*TR-60*	*TR-50*	*TR-40*	*TR-30*	*TR-20*
-17	-18	-21 (287)	-21 (1244)	-21 (992)	-21 (4671)	-20	-20	-17	-17
-19	-21 (2776)	-21 (5209)	-21 (2423)	-21 (1721)	-21 (5568)	-20	-19	-17	-16
-18	-20	-20	-21 (488)	-21 (611)	-21 (1668)	-19	-18	-17	-17
-18	-18	-21 (1711)	-21 (928)	-21 (1696)	-20	-20	-17	-18	-16
-19	-20	-20	-21 (345)	-21 (295)	-20	-20	-19	-18	-17

Data format: Maximum |fitness| (Generation number).

Table 2. Average run results of 5 iterations of PDB sequences after conversion into HP sequence; maximum generation = 6000

PDB ID	Length	*WT*	*TR-100*	*TR-90*	*TR-80*	*TR-70*	*TR-60*	*TR-50*
1PJF	46	-22	-24.6	-24.7	**-25**	-24.5	-24.5	-24
1AAF	55	-13.5	-13.6	**-15**	-14.5	-14.4	-14.3	-14
2PTL	78	-21	-22	-24.6	**-24.9**	24.8	-24.7	24.4
1GH1	90	-22.5	-26	-29	**-29.7**	-29.3	-28.5	-28
2GG1	102	-28.4	-31.8	-35	**35.5**	-34.3	-34.3	-34
2CQO	119	-37.5	-41	-44	**-44.5**	-44	-44	-40

Source: PDB sequences [43].

6000 generations was allocated for these particular series of experiments. PSP sequences [34], [38] shown in Table 1 and Table 2 for the 2D HP model [37]. Unlike Table 2, in Table 1, if during the iterations this optimal value was not reached, the maximum value achieved within generations is displayed. Fig. 3 shows the *Generation* vs. *Overall similarity* plot. In Fig. 3, it is shown that for the *WT* run, the *overall similarity* reached ≈80% very rapidly (around the 50th generation) from an

initial value of ≈35%, before stabilizing at ≈90% *similarity* after the 150th generation. This clearly supports (17) in that without any twin removal policy, the overall population quickly becomes strongly correlated and diversity is lost. The rapid nonlinear growth up to the 50th generation is also supported by AB_1 and AB_2 in (9) and (11) respectively, with AB_2 being the dominant component in the *overall similarity*, because each crossover is more successful in generating twins similar to their parents regardless of the crossover position, and also the biased selection procedure identified in (8) is embedded in AB_2. In the 5 separate iterations (Table 1) *WT* never reached the putative ground value and its maximum fitness stalled, generally before the 250th run, though the simulation ran for the entire 6000 generations. This is a direct consequence of twins with a higher fitness appearing in the population, thereby slowing the convergence over time as the population becomes less diverse. The *overall dissimilarity* or diversity in the chromosomes remained around 10%, which was insufficient to maintain a search capability, and so it became trapped due to premature convergence. It must be emphasized that with such a high number of generations the effect of mutation is negligible even if *elitism* is applied, as highlighted in Section 5**.** The elite population is clearly not deriving any benefit from the mutation operation. It is also clear that *TR-80* displays the best performance for correlated twins as the population maintains the most favorable balance between the *overall similarity* (chromosome *correlation)* so keeping the search stochastic to aid convergence, and upholding diversity by supporting the growth of dissimilar, but competent chromosomes. The generalized selection procedure delineated in Section 5 supports these newly created chromosomes as well as existing correlated chromosomes by ensuring the entire selection procedure is less biased.

7 Conclusion

The ease of Genetic Algorithm (GA) implementations has made them a popular solution for many optimization problems, with the expectation that they can be effectively and accurately applied to even complex optimization problems such *ab initio* protein structure prediction (PSP). This neglects however, the crucial role of the growth of similarity and chromosome twins has upon the population, which can lead to premature convergence. The twin problem can impair its performance ultimately leading to premature convergence or the stall condition. We have highlighted the fallacies within the selection procedure and shown the 'accrued benefit' from the crossover operation. A generalized schemata theorem has been proposed which highlights the need of twin removal and generalization of the schemata theorem for consistent GA performance. The definition of twins has been relaxed to not only embrace duplicate chromosomes, but also to take cognizance of strongly-correlated chromosomes. It has been observed [27], that while even in relatively simple landscapes, failure to remove twins can lead a GA frequently getting trapped in earlier generations. This problem has been overcome within the generalized framework presented in this paper, with *chromosome correlation factor* (CCF) setting to 0.8, affording the best performance.

References

1. Berger, B., Leighton, T.: Protein Folding in the Hydrophobic-Hydrophilic (HP) Model is NP-Complete. Journal of Computational Biology 5, 27 (1998)
2. Crescenzi, P., Goldman, D., Papadimitriou, C., Piccolboni, A., Yannakakis, M.: On the complexity of protein folding (extended abstract), p. 597. ACM, New York (1998)
3. Chen, M., Lin, K.Y.: Universal amplitude ratios for three-dimensional self-avoiding walks. Journal of Physics A: Mathematical and General 35, 1501 (2002)
4. Guttmann, A.J.: Self-avoiding walks in constrained and random geometries. Elsevier, Amsterdam (2005)
5. Jiang, T., Cui, Q., Shi, G., Ma, S.: Protein folding simulation of the hydrophobic-hydrophilic model by computing tabu search with genetic algorithms. In: ISMB (2003)
6. König, R., Dandekar, T.: Refined Genetic Algorithm Simulation to Model Proteins. Journal of Molecular Modeling 5 (1999)
7. Lamont, G.B., Merkie, L.D.: Toward effective polypeptide chain prediction with parallel fast messy genetic algorithms. In: Fogel, G., Corne, D. (eds.) Evolutionary Computation in Bioinformatics, p. 137 (2004)
8. Pedersen, J.T., Moult, J.: Ab initio protein folding simulations with genetic algorithms: simulations on the complete sequence of small proteins. Proteins 29, 179 (1997)
9. Schulze-Kremer, S.: Genetic Algorithms and Protein Folding, vol. 1996 (2007)
10. Takahashi, O., Kita, H., Kobayashi, S.: Protein Folding by A Hierarchical Genetic Algorithm. In: 4th Int. Symp. AROB (1999)
11. Unger, R., Moult, J.: On the Applicability of Genetic Algorithms to Protein Folding. In: The Twenty-Sixth Hawaii International Conference on System Sciences, p. 715 (1993)
12. Unger, R., Moult, J.: Genetic Algorithms for Protein Folding Simulations. Journal of Molecular Biology 231, 75 (1993)
13. Hoque, M.T., Chetty, M., Dooley, L.S.: Significance of Hybrid Evolutionary Computation for Ab Inito Protein Folding Prediction. Springer, Heidelberg (2006)
14. Hoque, M.T., Chetty, M., Dooley, L.S.: A New Guided Genetic Algorithm for 2D Hydrophobic-Hydrophilic Model to Predict Protein Folding. In: IEEE CEC. IEEE Computer Society Press, Los Alamitos (2005)
15. Bastolla, U., Frauenkron, H., Gerstner, E., Grassberger, P., Nadler, W.: Testing a new Monte Carlo Algorithm for Protein Folding. Nat. Center for Biotech. Info. 32, 52 (1998)
16. Liang, F., Wong, W.H.: Evolutionary Monte Carlo for protein folding simulations. J. Chem. Phys. 115 (2001)
17. Shmygelska, A., Hoos, H.H.: An ant colony optimization algorithm for the 2D and 3D hydrophobic polar protein folding problem. BMC Bioinformatics 6 (2005)
18. Fogel, D.B.: EVOLUTIONARY COMPUTATION Towards a new philosophy of Machine Intelligence. IEEE Press, Los Alamitos (2000)
19. Sareni, B., Krähenbühl, L., Nicolas, A.: Effective Genetic Algorithms for Solving Hard Constrained Optimization Problems. IEEE Transaction on Magetics 36 (2000)
20. Rudolph, G.: Convergence analysis of canonical genetic algorithms. ITNN 5, 96 (1994)
21. Altenberg, L.: The Schema Theorem and Price's Theorem Foundations of Genetic Algorithms 3 (1995)
22. Fogel, D.B., Ghozeil, A.: Schema processing, proportional selection, and the misallocation of trials in genetic algorithms. Information Science 122, 93 (2000)
23. Michalewicz, Z.: Genetic Algorithms + Data Structures = Evolution (1992)
24. Whitley, D.: An Overview of Evolutionary Algorithms. Journal of Information and Software Technology 43, 817 (2001)

25. Ronald, S.: Duplicate Genotypes in a Genetic algorithm. In: IEEE WCCI, p. 793. IEEE Computer Society Press, Los Alamitos (1998)
26. Haupt, R.L., Haupt, S.E.: Practical Genetic Algorithms (2004)
27. Hoque, M.T., Chetty, M., Dooley, L.S.: Critical Analysis of the Schemata Theorem: The Impact of Twins and the Effect in the Prediction of Protein Folding using Lattice Model, GSIT, MONASH University, TR-2005/8 (2005)
28. Deb, K., Goldberg, D.E.: An investigation of niche and species formation in genetic function optimization. In: 3rd Int. Conf. on Genetic Algorithms, p. 42 (1989)
29. Coello, C.A.C.: An Updated Survey of GA-Based Multiobjective Optimization Techniques. ACM Computing Surveys 32, 109 (2000)
30. Rogers, A., Prügle-Bennett, A.: Genetic Drift in Genetic Algorithm Selection Schemes. IEEE Transaction on Evolutionary Computation 3, 298 (1999)
31. Eshelman, L.J., Schaffer, J.D.: Preventing premature convergence in genetic algorithms by preventing incast. In: 4th Int. Conf. on Genetic Algorithms, p. 115 (1991)
32. Hoque, M.T., Chetty, M., Dooley, L.S.: Non-Isomorphic Coding in Lattice Model and its Impact for Protein Folding Prediction Using Genetic Algorithm. In: IEEE Computational Intelligence in Bioinformatics and Computational Biology IEEE CIBCB, Canada (2006)
33. Toma, L., Toma, S.: Contact interactions methods: A new Algorithm for Protein Folding Simulations. Protein Science 5, 147 (1996)
34. Lesh, N., Mitzenmacher, M., Whitesides, S.: A Complete and Effective Move Set for Simplified Protein Folding. In: RECOMB (2003)
35. Bornberg-Bauer, E.: Chain Growth Algorithms for HP-Type Lattice Proteins. In: RECOMB'97 (1997)
36. Backofen, R., Will, S.: A Constraint-Based Approach to Fast and Exact Structure Prediction in Three-Dimensional Protein Models. Kluwer Academic Publishers, Dordrecht (2005)
37. Dill, K.A.: Theory for the Folding and Stability of Globular Proteins. Bio-chemistry 24, 501 (1985)
38. Hart, W.E., Istrail, S.: HP Benchmarks (2005), http://www.cs.sandia.gov/
39. Vose, M.D.: The Simple Genetic Algorithm. MIT Press, Cambridge (1999)
40. Holland, J.H.: Adaptation in Natural And Artificial Systems. MIT Press, Cambridge (2001)
41. Goldberg, D.E.: Genetic Algorithm Search, Optimization, and Machine Learning. Addison-Wesley Publishing Company, Reading (1989)
42. Davis, L.: Handbook of Genetic Algorithm. VNR, New York (1991)
43. PDB, Protein Data Base, vol. 2007 (2007), http://www.rcsb.org/pdb/
44. Digalakis, J.G., Margaritis, K.G.: An experimental Study of Benchmarking Functions for Genetic Algorithms. Intern. J. Computer Math. 79, 403 (2002)