

# Enhancement of the Shifting Balance Genetic Algorithm for Highly Multimodal Problems

Jun Chen  
Computing and Information Science  
University of Guelph  
Guelph, Ont., N1G 2W1, Canada  
Email: jun@uoguelph.ca

Mark Wineberg  
Computing and Information Science  
University of Guelph  
Guelph, Ont., N1G 2W1, Canada  
Email: wineberg@cis.uoguelph.ca

**Abstract-** The Shifting Balance Genetic Algorithm (SBGA) is an extension of the Genetic Algorithm (GA) that was created to promote guided diversity to improve performance in highly multimodal environments. Based on a new behavioral model for the SBGA, various modifications are proposed: these include a mechanism for managing dynamic population sizes with population restarts, and communication among the colonies. The enhanced SBGA is compared against the original SBGA system and other multipopulation GA systems on a complex mathematical function (F8F2) and on the NP-complete 0/1 Knapsack problem. In all cases the enhanced SBGA outperformed all other systems, and on the 0/1 Knapsack problem it was the only one to find the global optimum.

## I. INTRODUCTION

While the GA has proven to be very effective, it can still get trapped at local optima, which is known in the GA literature as premature convergence. To a great degree lessening premature convergence can be seen as the problem of diversity maintenance. Many mechanisms have been developed to help the GA maintain diversity in a population, such as Restricting mating practice [1][2], adaptive mutation rate or selection pressure [3], random immigrant [4], restart / re-initialization mechanism [5] and multiple population solution [6] [7].

It would be useful if the diversity can be added in a guided fashion – less blindly. Blindly increasing the diversity can cause the GA to experience random drift in the search area and consequently lose its search efficiency. An algorithm has already been designed with these issues in mind: the Shifting Balance Genetic Algorithm (SBGA) [16]. The SBGA is a multi-population GA (similar to many parallel GAs), but with mechanisms introduced to promote directed diversity.

The SBGA when introduced was designed with user chosen parameters for both the number and size of the populations. These were chosen based on an initial model of how the SBGA behaved and was never investigated further. In this paper we propose a new analysis of the behavior of the SBGA and show that modifications inspired by the new analysis improve the SBGA's ability to handle multimodal environments.

## II. DIVERSITY AND ADAPTIVE POPULATION SIZES

### A. Maintaining Diversity

There are different ways measuring diversity, e.g. *pair-wise distances diversity*, *informational entropy*

*diversity*. Maintaining diversity usually can be done in two ways: either by decelerating the process of gene fixation or by reintroducing the diversity after it has been lost. Common diversity maintenance or enhancing techniques are increasing mutation, preventing inbreeding, decreasing selection pressure, restarting the populations, and the use of multiple populations.

One technique for introducing diversity is simply increasing mutation. Grefenstette [4] introduced the idea of *random immigrants* into the population. Here randomly generated individuals are inserted into the population to be a source of random crossover segments.

Inbreeding promotes diversity loss. Consequently, some restricted mating practices such as using species identity tag and inbreeding prohibitions are used [2]. These techniques can monitor individual gene makeup, fitness, or mating history.

Decreasing the selection pressure is also another way to help maintain diversity. There are many different ways to do this, the simplest being the reduction of the slope used in linear rank selection.

Re-initialization the population, in whole or in part according to some criteria or policy can reintroduce diversity after it has been lost. In practice the criteria used can include: cost bounds, time bounds, improvement probability bounds and convergence bounds, along with various combinations [5].

A multi-population GA (also called the Island Model) localizes members into sub-populations, called islands, where each island can potentially follow a different search trajectory through the search space and develop in more or less isolated niches [9].

### B. Adaptive Population Size

There has been very little work on adaptive population size schemes, despite the fact that the population size is a critical parameter in a GA and it largely determines the quality of the solution [10]. The limited amount of work done can be broken down into two groups: direct evolution of the population size parameter [14]; and mechanisms that indirectly affect the population size, which include GAVaPS [11], SOS [13] and BGA [12]. We will be using ideas from the last approach so we will go into further detail for that system.

Schilerkamp-Voose and Mühlenbein [12] use sub-populations competition that changes the sizes of sub-population in their breeder genetic algorithm (BGA). There the total number of individuals is fixed, whereas the size of a single group varies. In every generation

the subpopulations compete based on the fitness of the best of individual of the group. At regular intervals a quality criterion is used on the each subpopulation; the group with best quality increases the size and all other groups are decreased. A lower limit is used for the size of each group, so no population is lost. At other regular intervals, the best individual from each subpopulation emigrates to all populations and the best fitness of the groups becomes equal. We will borrow these ideas for the SBGA when we introduce subpopulation competition among colonies (a colony is a specialized subpopulation), although the mechanism will be slightly modified.

### III. SHIFTING BALANCE GENETIC ALGORITHM

#### A. Introduction

The Shifting Balance Genetic Algorithm was originally designed to use multiple populations to solve moving peak problems in dynamic environment; but it also can help GA as a function optimizer to overcome local optima or lessen the premature convergence in multimodal environments. Since this paper re-examines the behavior of the SBGA and improves on the algorithm, a brief overview of the system is presented below. For a more detailed explanation of the SBGA, see [15] [16].

#### B. Core group and Colonies

SBGA is a multiple population GA similar to the island model. But unlike other island model GAs it does not divide the population to several demes with the same size. In SBGA the populations are classified into two categories: A large central population, called the core and a series of smaller, helping population called colonies

The core is responsible for exploring the area of the search space that seems most promising and performing exploitation through receiving the elite immigrants from the colonies, while the colonies explore the areas of the search space where the core population does not yet have a presence. The core has a large population size so that it has an increased ability to do search or perform hill climbing since the more members the GA has, the more children it can produced and hence generate more trials in that area.

The colony is set to a smaller population size in order for each colony's gene make-up to more easily change, however this also can cause a reduced search power.

#### C. Selecting for Distance from the Core

For the SBGA to perform as desired, a mechanism is needed that allows a population to relocate to other places in the search space, thus escaping from a local maximum. The SBGA has a mechanism for determining whether an individual from the colony is searching in the same area as the core, which is implemented as a distance to the core: it is the average Hamming distance between a colony member and each member of the core.

The distance is used as a fitness function for the members in the colony when the colony becomes too close to the core. The members will then be selected for reproduction, not only according to their objective function value but also according to their distance from the core. They will *evolve* into a new area of the search space.

Since we intend the colony to follow the objective function's landscape, even as it moves away from the core, we are faced with a bi-objective optimization task. We handle this bi-objective optimization as follows:

The population for the next generation is split into two sections. The first section is selected under the objective function, and the second under the distance to the core. When filling a subsection, the parents are selected from the previous generation's entire population. However, the fitness function used during the selection is the one associated with the sub-population being filled.

In order to implement the above bi-objective approach to moving the colony away from the core, we need to determine the portion ( $\kappa$ ) of the population governed by the distant to the core versus the objective function. This is calculated using a formula called the *percentage similarity*, see [16] for details.

#### D. Migration from Colony to Core

In the SBGA, the colony sends members, called *migrants*, back to the core. During migration the colony may send all of its members to the core or only some portion thereof. The migrants are chosen from the elite members of the colony (25% of the colony has been used).

Since migration of the colony members disrupts the core group, time is given for the colony to evolve potentially useful members. The number of generations between migrations is called the *migration interval*. To reduce the pressure on the core even more, only a few colonies are allowed to send migrants in the same generation.

Just like all multiple-population based GAs, the SBGA needs a method to integrate the migrants arriving from the colony into the core's population. The host population is temporarily enlarged by the migrants and then pared down again to its normal size after reproduction. In other words, selection for reproduction is performed on the extended population, host members plus immigrants, but the new population will only be the size of the original population.

## IV. ANALYSIS OF THE BEHAVIOR OF SBGA

### A. Preliminary Experiments and Questions

In our description of the SBGA we assumed that the core population would behave as a regular GA enhanced by novel solutions found by the colonies. The large population of the core was assumed to be able to do the exploitation of an area with the small, more easily changed colonies providing the exploration. But

whether the SBGA does search in this way was never really proved. One of the ways to help us to understand the SBGA's behavior is to monitor each colony's portion governed by the distant to the core, i.e. its  $\kappa$  value.

We should observe the following: at first the core should repel all the colonies away, driving most of their  $\kappa$  value to nearly 0. Some time later if the current best colony is even better than core, immigrants entering the core will force the core to overlap with the colony, thereby forcing the colony's  $\kappa$  higher. Shortly after the repulsive force acted on the colony by the core will cause the colony to move away from the core and the  $\kappa$  value will drop. Maybe several generations later another colony becomes the best and the process will repeat.

Experiments were performed to see if this behavior actually happened. The  $\kappa$  values of all colonies were recorded during various runs of the SBGA using the same fitness function used to test the original SBGA system. Furthermore, a modified SBGA without the migration from the colonies to the core was used as a comparative system to the SBGA. The fitness function is the 6<sup>th</sup> dimensional F8F2 minimization problem (see [17]). The definition of F8F2 function is

$$F8F2(\mathbf{x}) = \sum_{i=1}^6 \left( 1 + \frac{(F2(\mathbf{x}))^2}{4000} - \cos(F2(\mathbf{x})) \right) \quad (1)$$

$$F2(\mathbf{x}) = 100(x_i^2 - x_{i+1}) + (1 - x_i)^2$$

The parameter settings used for the experiment are summarized in **Table 1**. Each experiment lasted 300 generations and was repeated 60 times.

In these 60 experiments we find the trend of the  $\kappa$  value is very similar. **Fig. 1** shows a typical result.

**Table 1.** SBGA parameter setting

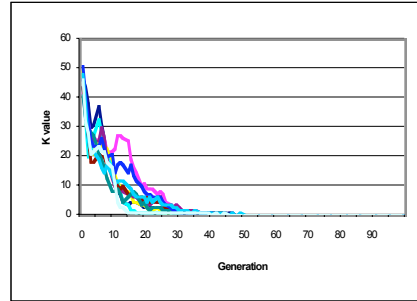
<b>Population Size</b>		<b>Selection</b>	
Core	1000	Linear rank selection	
Colony	100	Low=0.0 (high=2.0)	Elitism
<b>Core</b>		<b>Colonies</b>	
Mutation rate	0.006	Mutation rate	0.01
(bits/locus)		(bits/locus)	
Prob. of crossover	0.7	Prob. of crossover	0.9
One point crossover for		Migration size	25% colony
both Core and Colony		Migration interval to core	6
		Number of Colonies	10

In Fig. 1a where no immigration occurs, the colonies will soon be repelled away to other areas of the search space and no longer overlap the core. After 15-40 generations all the  $\kappa$  values for colonies are almost 0, which means the genetic composition of each colony becomes completely dissimilar to that of the core. Now look at Fig. 1b: when the colonies send immigrants to core, the  $\kappa$  values remain high over the entire course of the experiment for most of the colonies. We do not see individual colonies rise and decay sequentially as the core becomes interested in that colony's area. Rather the core seems to overlap with almost all the colonies

simultaneously. This seems to mean that the core is trying to repel all of the colonies away at the same time, although the repulsive force is not constant for each of them.

Based on the result of the preliminary experiments, it seems the SBGA does not always work as originally expected. Perhaps there is a better explanation search behavior of the SBGA that allows it to better overcome local optima. In the following section we present a new hypothesis of the core's role in search.

#### a) No immigration from colony to core



#### b) With migration from colony to core

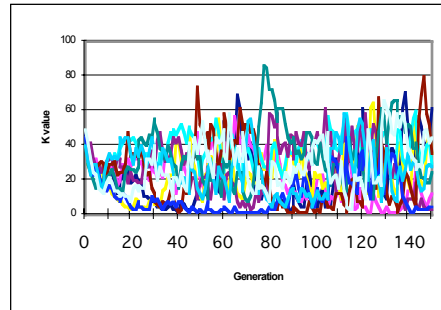


Figure 1. Overlap between core and colonies (dynamic  $\kappa$ )

#### B. New Analysis of the Core's Role for Search

In the above experiment the core overlaps with all of them throughout the experiment. One possible explanation is that, because of the core's large population size and the fact that the immigrants are the elite members; the immigrants from all colonies are entering the core and staying around long enough to create multiple niches in the core. This causes the core to become disorganized, which fragments core's gene pool and confuses the core on where in gene space to go. If this is the case, the core can be seen as a gathering center for the different immigrants from the colonies. If most of colonies continue to find better solutions and contributes them to the core, the core actually will not be stable and cannot focus on searching in just one area of the gene space.

However, the original experiments SBGA really do work better than GA when test on some multimodal environments [15] [16]. What makes it better? One reason probably comes from simply using multi-

populations. Another reason may be that the main power does not come from the core moving to where some colony is lying in the search space and just repelling it away and then exploring the area by itself; rather the mechanism kicks in at the beginning when the core tries to repel all the colonies away. This kind of repulsive force can make the colonies more active and search more areas, thus increasing the probability that one of them finds a better local optimum or even the global optimum.

From this point of view, the key role of the core is not seen as that of an exploiter who mines the area found by the current best colony, but rather more like the center of a repulsive force or possibly even a memory center who receives immigrants from all the colonies and tell them “you have already looked here, you can search some other place”. The core may even be better than a simple memory since it improves on what was stored by evolving it.

As a natural consequence of this new explanation, it becomes feasible to redesign the SBGA and improve its search ability and robustness. In next section we will consider how to redesign the SBGA in light of our new understanding.

## V. REDESIGNING THE SBGA

### A. Introduction

Through the analysis from the previous chapter we questioned the explanation of the behavior of the original SBGA and supplied some alternative explanations on why the SBGA is better than simple GA, especially when facing multimodal fitness landscapes. Based on this analysis, we redesign the SBGA along three lines, which are presented below.

### B. Small Core or Big Core

According to the analysis in section 4 the core cannot be easily shifted into the area of the best colony; furthermore if it did manage to cover the new territory, it would not efficiently search it. Therefore, a big core size is probably a waste of resource. As a result in the modified SBGA we decided to shrink the core size and give more search labor to do exploration by increasing the colony number.

However, we must not choose a core size that is too small, for it should be large enough to repel all of the colonies simultaneously. Consequently we decided to reduce the core from 1000 members to 200 members.

### C. Dynamically modifying the Colonies' population sizes

If the core's population size is decreased, then the colonies population must increase. The new analysis of the SBGA's behavior leads us to the conclusion that it is the colonies and not the core that is responsible for exploiting a region in gene space. Hence, we could give the colony with the current highest performance the greater population size giving it the greater search power. Thus, we introduce competition among the

colonies directly: a better colony will be given more members so the population size of the colony is adapted according to their performance.

In section 2 we saw that Schilerkamp-Voose and Mühlenbein added subpopulation competition for population size to their breeder genetic algorithm [[12]]. We will add that mechanism, with modifications, to the SBGA to determine which colony should have the greater search power.

Each colony will be evaluated over some generation interval (e.g. 5 generations). The colony with the best performance will have their population size increased, and likewise all other colonies will have their population size decreased.

The first step is the choice of which population is to have its size increased. This part of the algorithm is where we differ from the BGA the most. We removed their complex mechanism for “averaging” the subpopulation's performance over the generations. We also removed the evening-out of the best of a population through broadcasting each population's elite member to all other populations. In their place, two criteria are used to select which population should have its size increased:

1. *Rank Value*
  - Colonies with a higher fitness of its best member get a higher priority of consideration
2. *Colony Stagnation*
  - We consider a colony stagnant if there is no improvement in last 10 generations

Expanding on the second criterion, the chosen colony needs to show it is not stagnant before it can be chosen. This means each colony will be monitored to see when the colony last improved. The improvement measure is based on the best fitness of individual in the population, not the average fitness of the population. So the generation gap between the current generation and the generation when the best fitness was found is calculated, and is called the *stagnation interval*. If this gap is bigger than a user defined parameter or threshold (we used the value of 10 to match the BGA *quality criterion*, which our mechanism replaces) the colony will not be chosen and the algorithm will automatically turn to next one on the fitness list. If all the colonies are not eligible, the algorithm will randomly choose one. If there is a tie in the ranks for both criteria, a colony is randomly chosen from among the ties.

After one of the colonies is selected, all other colonies give a fixed percentage of their members (Schilerkamp-Voose and Mühlenbein used the value of 1/8 for their experiments) to the best non-stagnant colony. This is called the *gain criterion*.

### D. Restarting Dead Colonies

Since the colonies is now seen as the focus of the both exploration and exploitation, colonies that are not productive are more deleterious than previously thought and should be replaced. When this happens,

these colonies will restart using members from other colonies as seeds.

Three criteria are used for choosing the colony to restart:

1. Choose the lowest ranking colony.
2. Choose a colony with no improvements over the last few generations.
  - e.g. 10 gen. (from preliminary experiments)
3. Choose a colony with a diversity that is lower than some threshold.
  - e.g. 0.15 (from preliminary experiments)

Diversity is computed using the average information entropy at a locus across the population.

After the stagnant colony is chosen, we will restart it. The whole population will be re-initialized based on a seed individual from the colony that is having its population size increased. This seed individual, which will be the best individual from the donating population, will be used to create all of the individuals in the new population by performing bit mutation with a given probability (e.g. 30%). So the new population can be seen as a hyper-sphere around the seed in genotype space, with the radius of the hyper-sphere is decided by the probability of mutation.

#### *E. Communication among the Colonies*

By sharing information between colonies, colonies that are trapped at a local optimum can escape from the area. Even if the repelling force of the core is in effect, it will take time to rebuild the diversity of the colony. Migration from other colonies can prevent that from happening if this worse case scenario occurs.

Of course this kind of communication should not occur too frequently. Colonies, being small, can easily converge on immigrants from other colonies. The repelling force is a more gentle mechanism and so is preferable over the more harsh method of migration, so migration should only be used if the repelling force fails.

Immigration among multiple populations has been extensively studied for parallel GAs. Consequently, a lot of different topologies have been proposed and can be chosen from. In the modified SBGA, the Bi-directional ring topology is chosen. The Bi-directional ring is the simplest topology, and it also restricts the amount of immigrants flowing from colony to colony, which is vital, as we shall see shortly.

In the Bi-directional ring all the colonies are given two direct neighbors. In some generation interval (e.g.20), for each colony, the best 10 percent of the population are sent to its two direct neighbors; at same time the immigrants from two direct neighbors will also replace the worst 20 percent of this colony.

In the section 4, we claim that the migrants from all the colonies can fragment the gene pool of the core. If we introduce the bi-directional ring communication among the colonies, can the colony have the same

fragment problem? We don't believe that this is the case for the following reason.

Unlike the core, which needs to deal with all the colonies at the same time, the bi-directional ring constrains a colony so that it can only get members from its two direct neighbors instead of everyone. Consequently it has less chance of being fragmented and also has the opportunity to develop some genetic similarity with its neighbors. Although the bi-directional ring connects all of the colonies, immigrants from distant neighbors have to cross all the colonies in-between. Consequently, the gene makeup of the migrants will change as they migrate from population to population becoming more similar to the members of the distant colony. So this kind of migration is more smooth and gentle for each colony. In fact there is an existing natural example where this can be observed: ring species, such as Gulls around the Arctic Ocean [18] or Salamanders around desert of the San Joaquin Valley in California [19].

## VI. EXPERIMENTAL RESULTS AND ANALYSIS

### *A. Comparison among six different systems*

In the following experiment, we will see if further improvements can be observed when dynamic population size and communication among the colonies are introduced to the SBGA system. We will also turn on and off “repulsive central force” mechanism to see how effective it is.

There are 6 evolution systems that we compare: **1)** Original SBGA with a smaller core (**S**), **2)** SBGA with subpopulation competition for population size along with “dead colony” restart (**dynS**), **3)** SBGA with colony communication using a bi-directional ring (**comS**), **4)** System 1 without the “repulsive central force” – i.e. a multi-populational GA using isolated subpopulations with a “central clearing house” (**centerG**), **5)** System 3 without the “repulsive central force” – i.e. a multi-populational GA with a bidirectional ring topology and a “central clearing house” (**comG**), **6)** a simple GA with a large population (**G**). System 1 is based on the observation in section 5b (small core, many colonies), system 2 is based on the combination of 5c (dynamic population sizes among colonies) and 5d (colony restarts when non-productive) – the two were designed to work together, System 3 is based on section 5e (communication among colonies) and adds a bi-directional ring topology to the basic SBGA system. System 4 and 5 are respectively the counterpart of system 1 and 3, but with the repulsive central force turned off. This produce multi-

TABLE 2. Parameter settings the 6 systems

<b>Population Size</b>	<b>Selection</b>	
GA 2000, Core 200, Colony 100	Linear rank selection (max = 2), Elitism	
<b>GA and Core</b>	<b>Colonies</b>	
Mutation rate = 0.006 bits/locus	Mutation rate = 0.01 bits/locus, Prob. of xover = 0.9	
Prob. of xover = 0.75	Number of Colonies 18	
<b>Migration</b>	<b>Special for SBGA with competition</b>	
Size to core = 25 colony members	Colony size	Intervals
Size to colony = 10% colony	Initial: 100	Evaluation: 5
Migration interval: to core = 6, to colony = 20	Min: 40	Stagnation: 10
	Competition begins	30
	Lowest diversity	0.15

populational systems, the first with the colonies independent of each other, but sending members to a central location; the second is analogous to an island model ring system with the addition of the central “clearing house”.

For all 6 systems the parameter setting used are listed in TABLE 2. Dimension 4, 6 and 8 of F8F2 function are used to test on these 6 evolutionary systems. The total population size for all systems is all fixed to 2000. Each experiment lasted 300 generations and was repeated 60 times.

Table 3 gives the results for the 6 systems using F8F2 with 4, 6 and 8 dimensions.

The results are given in the form of the median of the 60 runs with confidence bounds. We record the median instead of the mean because the results are not normally distributed. We are testing for statistical significance of the comparison between the 6 systems. If the upper and lower bounds don’t cross, neither system is better than the other. This test is not as sensitive as using ANOVA with pair-wise Student’s T tests. These were done (with a confidence level of 95%) and they confirm the results presented here. The upper and lower percentiles are obtained from the Thompson - Savur method for finding confidence intervals around a median, using a Bonforonni correction (we performed 97 independent comparisons in all our experiments; we therefore always divide our  $\alpha$  by 97).

Using the above table to order the systems by performance we get (with statistical significance<sup>1</sup>):

Dim 4: dynS > comS > S ≥ comG ≥ centerG > G

Dim 6: dynS | comS > S > comG | centerG > G

Dim 8: dynS | comS > comG | S | centerG > G

From the result we can see for F8F2 problem on dimension 4, 6 and 8, the SBGA with the modifications

TABLE 3: Comparison among the 6 GA systems

System	Lower Bound 29 %tile	Median	Upper Bound 71 %tile
Dimension 4			
G	0.04856	0.08917	0.11312
centerG	0.01052	0.02070	0.04317
comG	0.01009	0.01992	0.03955
S	0.00995	0.01018	0.02060
comS	0.00012	0.00989	0.01057
dynS	0.00000	0.00000	0.00147
Dimension 6			
G	0.12927	0.14611	0.18225
centerG	0.06133	0.08914	0.11941
comG	0.05982	0.08991	0.13231
S	0.02228	0.05994	0.08895
comS	0.02093	0.04099	0.05960
dynS	0.01003	0.01444	0.03319
Dimension 8			
G	0.20164	0.26798	0.30194
centerG	0.13455	0.18464	0.22640
comG	0.11260	0.15798	0.19805
S	0.13113	0.15580	0.21074
comS	0.03590	0.06423	0.12136
dynS	0.04199	0.07654	0.13685

based on our analysis are better than any of the other systems.

#### B. Modified SBGA on the 0/1 knapsack problem

In the previous experiment, the two modified versions of the SBGA have demonstrated an improved ability to escape from local optima on a theoretical optimization function, F8F2. However, although the F8F2 function is carefully chosen to satisfy our test purposes, it would be interesting to test the modified systems on a more “real-world problem” such as an NP-hard problem, to see whether the improvements can still be seen.

#### Problem Definition:

The 0/1 knapsack problem is an NP-hard constraint problem yet is easy to implement. The knapsack problem can be simply described: given a set of items, each with a cost and a value, determine the number of each item to be included in a collection such that the total cost is less than some target cost and the total value is as large as possible (The 0/1 knapsack problem restricts the number of each items to zero or one).

The task can also be formulated as: for a given set of weights  $W[i]$ , profit  $P[i]$ , and capacity  $C$ , find a binary

<sup>1</sup> Here ‘A>B’ means system A is better than system B with statistical significance and ‘A | B’ is means that system A is statistically indistinguishable from system B. ‘A≥B≥C’ means that system A cannot be statistically distinguished with B and system B cannot be distinguished with system C but the system A is statistically better than system C

vector  $x = \langle x[1], \dots, x[n] \rangle$ , such that  $\sum_{i=1}^n x[i] \cdot W[i] \leq C$  and for which  $\rho(x) = \sum_{i=1}^n x[i] \cdot P[i]$  is maximal.

*Test Data:*

In the following test, we don't randomly create the test data by ourselves since it is hard to know where the real the global optimum lies. The data we used has been downloaded from a website and item number  $n=200$ , Maximal capacity  $C=5000000$  and the Maximal optimal profit  $r(x) = 5180258$  (the global optima). Figure 4 gives the relationship between profit value and weight.

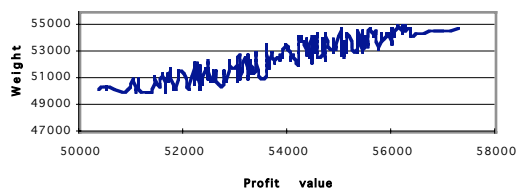


Figure 2: The relation btw the item weight and item profit value

From **Figure 2** we can see that there is a rough correlation between the profit value of an item and its weight. But the mapping is not completely linear as there is a lot of noise. Consequently, the profit value and weight are weakly related. For the actual profit/weight values, contact the authors.

*Dealing with Constraint:*

For the constraint handling technique we used chromosomal repair using a greedy algorithm. All items in the knapsack are sorted in decreasing order of their profit to weight ratios. The procedure always chooses from the last item (the smallest profit to weight ratios) for deletion and recursively removes the item from the knapsack until the individual solution is feasible.

*Parameter Setting:*

The 6 systems in last section are still used to do the comparison, with different population sizes. Along with the 6 systems, we will also test the original big core SBGA. The big core SBGA has 6 colonies, population size for each colony and the core are respectively 50 and 350; all other parameter settings are the same. Most of the parameter setting are the same as in TABLE 2, with the differences given in TABLE 4.

TABLE 4. Parameter settings differences from TABLE 2

<b>Population Size</b>	GA 650, Core 150, Colony 50
<b>Colonies</b>	Number of Colonies 10
<b>Migration</b>	Size to core = 25% of colony
<b>Special for dynS</b>	Colony size: Initial - 50, Min - 20 Migration size to Core 10 (fixed)

For all the systems the total population size is 650. The reason for this decrease from 2000 to 650 is that preliminary tests show that the GA can perform nearly as well with this population size, and we can generate the experiments faster.

*Experiment result:*

Each experiment lasted 500 generations and was repeated 60 times.. Table 5 gives the result.

TABLE 5: Comparison among the 7 GA systems

System	Lower Bound 29 <sup>th</sup> %tile	Median	Upper Bound 71 <sup>st</sup> %tile
S	5179415	5179522	5179706
centerG	5179382	5179560	5179712
bigS	5179399	5179560	5179733
comG	5179576	5179716	5179765
G	5179537	5179784	5179916
comS	5179799	5179913	5179998
dynS	5179851	5179944	5180026

Performing a pair-wise comparison from the table we obtain the follow statistical relations:

$$\text{dynS} \mid \text{comS} > G \geq \text{comG} \geq S \mid \text{bigS} \mid \text{centerG}$$

Consequently, we see that the two modified systems are better than other 5 systems. In fact, the SBGA with competition is the only system that found the global optimal; it even found it twice. But the SBGA with the small core doesn't work very well, contrary to expectations. This enigma may be resolved if you notice that the single population GA is better than both small and big core SBGA and the central island model (with no colony communication) and is indistinguishable from the island model with colony communication (which allows the system to behave more as a unified whole). This leads us to believe that the problem isn't as multi-modal as we expected. For F8F2 problem there are innumerable local optima, so more colonies is probably more helpful and for 0/1 knap problem we don't know what kind of fitness landscape it is and form the result of experiment the single big population seem even better than normal island model.

However for the two modified version of the SBGA, the systems has the ability to dynamically allocate population size so when the problem is more suitable for a big single population, the two new systems can, to some degree, automatically adjust them to it. Consequently, the two modified versions of SBGA are more robust them the original one.

VII. CONCLUSIONS

The SBGA was originally created with a central main population, the core, that performed the main search, flanked by small populations, called colonies, that

performed exploration. The colonies would inform the core of their findings through immigration. A mechanism, called the “repulsive central force” was devised to keep the colonies from searching in the same area as the core.

It was envisioned that one of the colonies would find a promising area, inform the core, then get pushed into a new area as the core started to explore the area the colony had found. However, on further study the repulsive central force was acting simultaneously, not sequentially on the colonies. Furthermore it was discovered that a small core with many colonies worked better than the large core with few colonies.

These facts prompted a new analysis to be devised. Now the colonies are seen as the primary source of exploration and exploitation; the core instead of being responsible for search, behaves more as a “global agitator” for the colonies, keeping them from staying too long in one place, while still allowing them to follow the fitness landscape.

This analysis leads to various new mechanisms to be added to the SBGA. Since the colonies are the centers of search, more care should be used in managing them. Consequently, the colonies were provided with dynamic population size so the colony that was searching the most promising area would get more members to search with. Colonies that were stagnant for too long are now seen as too detrimental and so were dissolved and restarted. Finally, migration was added between colonies as an alternate mechanism to prevent premature convergence.

These new systems were compared against the original SBGA as well as a large standard GA and various simple multi-population GAs on a hard multi-modal mathematical function (F8F2) and the 0/1 Knapsack problem. In all cases, the enhanced SBGA system excelled demonstrating its improved ability to handle highly multi-modal systems.

#### REFERENCE

- [1] Eshelman, L.J., Schaffer J.D. (1991). Preventing Premature Convergence in Genetic Algorithms by Preventing Incest. 4<sup>th</sup> ICGA, pp.115-112.
- [2] Craighurst, R., Martin, W. (1995) Enhancing GA Performance through Crossover Prohibitions Based on Ancestry. 6th ICGA, pp.130-135
- [3] T. Bäck, (1992). The interaction of mutation rate, selection, and self-adaptation within a genetic algorithm. PPSN 2.
- [4] J.J. Grefenstette. (1992) GAs for changing environments. PPSN 2, pp.137-144.
- [5] Eshelman, L.J., (1991) The CHC Adaptive Search Algorithm: How to Have Safe Search When Engaging in Nontraditional Genetic recombination. *Foundation of Genetic Algorithms*. pp. 265-283.
- [6] E. Cantu-Paz. (1998) A survey of parallel genetic algorithms. *Calculateurs paralleles, Reseaux et Systems Repartis*, 10(2): 141-171.
- [7] E. Cantu-Paz (2000). *Efficient and Accurate Parallel Genetic Algorithms*. Kluwer.
- [8] D. Whitley, S. Rana and R. B. Heckendorn (1998) *The Island Model Genetic Algorithm: On Separability, Population Size and Convergence*, Proceedings of the AISB Workshop on Evolutionary Computation.
- [9] Munetomo, M., Takai, Y., & Sato, Y. (1993). An efficient migration scheme for subpopulation-based asynchronously parallel genetic algorithms. 5th ICGA. pp. 649.
- [10] Harik, G., Cantu-Paz, e., Goldberg, D., & Miller, B. L. (1997). The gambler’s ruin problem, genetic algorithms, and the sizing of populations. In the 4th ICEC.. pp. 7-12
- [11] Michalewicz, Z. (1994) *Genetic Algorithms + Data Structures = Evolution Programs* (2nd edition). Springer-Verlag.
- [12] Dirk Schlierkamp-Voosen and Heinz Mühlenbein (1994). Strategy adaptation by competing subpopulations. PPSN 3, pages 199-208.
- [13] Jürgen Branke. (2002). *Evolutionary Optimization In Dynamic Environments* University of Karlsruhe, Germany. Kluwer Academic Publishers
- [14] Shisanu Tongchimaand and Prabhas Chongstitvatana. (2002). *Parallel Genetic Algorithm with Parameter control*.
- [15] M. Wineberg and F. Oppacher (2000). Enhancing the GA’s Ability to Cope with Dynamic Environments. In GECCO-2000. pp. 3-10.
- [16] M. Wineberg (2000). *Improving the behavior of the genetic algorithm in a dynamic environment*. PHD Dissertation, Carleton University, Canada.
- [17] Whitley, D., K. Mathias, et al. (1996). “Evaluating Evolutionary Algorithms”. *Artificial Intelligence* 85: 245-276.
- [18] Welty, Joel Carl. *The Life of Birds*, 3rd edition, 1982. NY: Saunders College Publishing. pg. 618.
- [19] Moritz C., C.J. Schneider, and D.B. Wake. 1992. Evolutionary relationships within the *Ensatina eschscholtzii* complex confirm the ring species interpretation. *Syst. Biol.* 41(3):273-291