



Foo, Y. W., Goh, C. S., and Li, Y. (2016) Speciation and Diversity Balance for Genetic Algorithms and Application to Structural Neural Network Learning. In: IJCNN 2016: IEEE World Congress on Computational Intelligence, Vancouver, Canada, 24-29 July 2016.

There may be differences between this version and the published version. You are advised to consult the publisher's version if you wish to cite from it.

<http://eprints.gla.ac.uk/118964/>

Deposited on: 05 May 2016

Enlighten – Research publications by members of the University of Glasgow
<http://eprints.gla.ac.uk>

Speciation and Diversity Balance for Genetic Algorithms and Application to Structural Neural Network Learning

Yong Wee Foo^{1,2}, Cindy Goh¹, Yun Li¹

¹School of Engineering, University of Glasgow, Glasgow, U.K.

²School of Engineering, Nanyang Polytechnic, Singapore

Foo_Yong_Wee@nyp.edu.sg; Cindy.Goh@glasgow.ac.uk; Yun.Li@glasgow.ac.uk

Abstract— Following analyzing existing challenges in addressing the balance between exploration and exploitation encountered by evolutionary algorithms, this paper develops a Genetic Algorithm with speciation (GASP). It first incorporates a novel encoding scheme and recombination method for a balanced genetic divergence when locating global optima in complex applications, such as structural and dynamic design of an artificial neural network (NN). GASP also addresses the problem of defining a measure and track population diversity whose NN structure is subjected to continual reorganization during the evolution process. Further, a novel approach to the neural network phenotype is developed, which maps it to a distinct genome with a variable length capable of fully representing the multilayer feed-forward NN structure. Using the concept generalized from linguistic complexity, the distance between strings can thus be derived from the single string and substring counts. The GASP is then applied to an NN design problem to forecast the energy consumption of a built environment. With the optimal NN structure, diversity is tracked and improved. The results show that the GASP succeeds in obtaining excellent accuracy and speed.

Keywords—genetic algorithm, neural networks, evolutionary computing, diversity, speciation

I. INTRODUCTION

Genetic algorithm (GA) is a heuristic search algorithm inspired by the Darwinian principle of evolution and natural selection. A GA based design approach centers on the concept of heredity to produce the ‘best of breed’ solutions to complex problems. In addition to design, the GA has been successfully applied to many other real-world problems such as task scheduling, resource allocation, telecommunication network routing, finance strategy planning, computer gaming and machine learning.

In a GA, possible solutions to a problem at hand are first encoded into strings or chromosomes. A fitness function is defined to measure the performance standing of the chromosomes. The objective of the GA is hence to search for the fittest chromosomes in a population of potential chromosomes through heuristic propagation of good genes from generation to generation in an evolutionary process. The hereditary nature directs healthy

parents to pass down good genes to the offspring with appropriate selection and reproduction mechanism.

Using a GA to optimize an artificial neural network (NN) allows solving NN problems more globally, as structural tuning and dynamic changes can be accommodated. However, maintaining an appropriate balance between exploitative and explorative characteristics is key to the success of evolutionary based approach [1].

In this paper, we develop a robust GA to design a globally optimal NN in a multimodal domain so as to enhance forecast accuracy and reduces structural inefficiency. We incorporate the GA with speciation (GASP) using a novel encoding scheme and recombination method with genetic divergence in locating global optima. We also develop a metric, using a concept generalized from linguistic complexity to measure and track the population diversity, upon analyzing the algorithm’s performance. The GASP will be implemented also to solve partially connected NN problems, for a better generalization at a reduced cost.

Section II reviews the related work in the area of reducing diversity drift in genetic search. Section III develops the GASP approach in detail. Section IV discuss the application of GASP in NN design. Section V examines the test results and Section VI provides conclusions and discusses future work.

II. EXISTING CHALLENGES IN BALANCING EXPLORATION VS EXPLOITATION

Complex optimization problem often lead to multimodal domains with multiple optima, global and local, exist in the search space. A standard GA (SGA), though advantageous as a global search technique, has its limitation in explorative search if the population is taken over by the “best individual” too quickly. This occurs when the search is directed to a local optima too aggressively with no new search space explored. As such, the genetic process is said to have converged prematurely with elements in the gene pool becoming similar to each other and further iterations of reproduction and mutation operations are unable to create new chromosomes. Genetic divergence can support global exploration and help to locate the global optima [2]. Hence many of the methods

presented in recent research for optimization of single-objective, multimodal functions aims towards reducing the effect of genetic drift.

A. Niching, Speciation and Crowding

Niching or speciation method mimics the natural evolution of different species thriving within an ecosystem occupying a niche area of their own. Species with similar biological features interbreed and competes to survive. In the physical environment where they dwell, resources are limited which must be shared among the species. In analogy, a niche is an optimum of the domain or attractor region in the search landscape, species are subpopulation while fitness are shared resource. Speciation in GA separates individuals into subpopulations of similar characteristics. These subpopulations are formed to investigate many peaks in parallel in the search domain [3]. In another words, they prevent GA from being trapped in local optima. The technique widely used in niching is called fitness sharing [4][5]. Fitness sharing reduces densely populated region in the search landscape by lowering the individual fitness level in this region due to diminishing resources. In a steady state, search moves across unexplored regions through fostering of other subpopulations where resources are more attractive. A sharing function measures the population diversity or similarities between the individuals based on either genotypic or phenotypic characteristics. Genotypic similarity, defined by a niche radius, is usually agreed by comparing the hamming distance between individuals via their genomic codes whereas phenotypic similarity is agreed by comparing values linked to physical parameters or topology such as the Euclidian distance. Two individuals closer than the niche radius are considered to be in the same species and hence share their fitness value. Phenotypic diversity based measure is preferred due to the slightly better results than genotypic diversity based measure [3][6].

Instead of dealing with an explicit radius, crowding attempts to replaces most similar individuals randomly taken from a subpopulation or crowd of size determine by a crowding factor [7]. The effect is to try to replace individual that is very similar with new individual thus preserving diversity. In another crowding technique called deterministic crowding, unlike the standing crowding whose competition is random in nature, the offspring competes directly with the parents and replaces the parents if its fitness is higher. This method presupposes that the genetic make-up of the parents would be nearest to the offspring [8]. And in probabilistic crowding, individuals with lower fitness are given a better chance to survive [9].

Other adaptations of the speciation techniques have also been highlighted and discussed in [12]. For instance, [13] used conservation of species technique to preserve its genetic information from one generation to another. [14] suggested adopting species-wise evolution technique to independently evolve these solutions to converge to their respective optima. [15] proposed fitness sharing to be applied within species rather than the entire population.

B. Diversity Measure

Population diversity is a key measurement in the above implementation of multimodal search techniques. It is also beneficial to measure and track the population diversity to better understand how the search is performed. Several methods for determining diversity both in the genotypic and phenotypic space have been used.

A common genotypic measure is pair-wise Hamming distance. Given P strings of length L , its pair-wise Hamming distance [6] is defined as:

$$H = \sum_{j=1}^{P-1} \sum_{j'=j+1}^P \left(\sum_{i=0}^{L-1} |y_{ij} - y_{ij'}| \right) \quad (1)$$

where $y_{ij}, y_{ij'} \in \{0,1\}$ and the generalized notation,

$$\sum_{k=1}^{M-1} \sum_{k'=k+1}^M f(x_k, x_{k'}) \quad (2)$$

is the sum of the results of the application of $f(x_k, x_{k'})$ to all pair-wise combinations the numbers x_k and $x_{k'}$ of given population of size M .

Other metrics used to evaluate population diversity in GA are gene-level entropy metric, chromosome-level neighbor metric and population-level centre of mass metric [10]. The formula for entropy, which is a measures of the degree of disorder, is defined as,

$$H_j = - \sum_{i=1}^N p_{ij} * \log_2 p_{ij} \quad (3)$$

where p_{ij} denotes the probability of an event a_{ij} occurring, and N the population size. At the gene-level for a population, Eq. (3) can be expressed as,

$$H(P(0)) = \frac{1}{l} \sum_{j=1}^l H_j \quad (4)$$

where H_j corresponds to the entropy in Eq. (3) for locus j of the whole population and l is the length of the chromosome.

The chromosome-level neighbor metric measures diversity at chromosome-level. The metric is defined using the Hamming distance with neighborhood. For the neighbors of chromosomes C_m at a distance equal to k , we have,

$$B_k(C_m) = \{C_j \in \{0,1\}^l \mid H = k\} \quad (5)$$

where $k \in \{0,1, \dots, l\}$, and H is the Hamming distance expressed in Eq. (1). We then make C_m a pivot to find the set of neighbors at each distance k . For the entire population, we can derive the chromosome-level diversity expressed as,

$$N(C_m) = \frac{\sum_{k=0}^l |B_k(C_m)| * k}{\sum_{k=0}^l |B_k(C_m)|} \quad (6)$$

where $|B_k(C_m)|$ is the cardinality of $B_k(C_m)$, that is, the number of neighbors of C_m at a distance k in the whole population.

In population-level centre mass metric [11], the population is treated as a matrix of genes. The centre of mass of a gene is taken with respect to an origin (0,0). To find the x coordinate of the centre of mass of a gene with binary value '1', we have the expression,

$$\bar{x}_1 = \frac{\sum_{i=1}^N \sum_{j=1}^l c(a_{i,j})}{\sum_{i=1}^N \sum_{j=1}^l a_{i,j}} \quad (7)$$

where $C(a_{i,j})$ is the column position j of gene $a_{i,j}$ where the gene has binary value ‘1’, and $\sum_{i=1}^N \sum_{j=1}^l a_{i,j}$ is the number of those genes where $a_{i,j} = 1$. Similarly to obtain the y coordinate for a gene with binary value ‘1’ is,

$$\bar{y} = \frac{\sum_{i=1}^N \sum_{j=1}^l R(a_{i,j})}{\sum_{i=1}^N \sum_{j=1}^l a_{i,j}} \quad (8)$$

where $R(a_{i,j})$ is the row position i of gene $a_{i,j}$ where the gene has binary value ‘1’, and $\sum_{i=1}^N \sum_{j=1}^l a_{i,j}$ is the number of those genes where $a_{i,j} = 1$.

In [20], we found the idea of defining a measure of diversity using genomes with variable length and structure interesting. This is discussed further in the next section.

III. SPECIATED GA

Although various literature has shown good results with niching and speciation techniques, our study shows that the technique should be adapted to best suit the problem definition. In NN optimization, we have designed a simple yet effective speciation approach by categorizing the NNs into their respective subpopulations based on the number of hidden nodes. The subpopulation are domains of optima and individuals within the species compete for survival through selection and intra-species crossover.

In a SGA, the process of crossover of individuals with different topology, might end up the child not having comparable topology causing a “loss” of structural innovation. As a result, these individuals are unlikely to survive in a global competition. SGA would then converged quickly with identical individuals dominating the search space.

In GASP, the intent of speciation is to protect structural innovation as these innovative structure would be isolated within their own species and hence a chance to be optimized. However, unlike standard niching implemented with a niching radius, we do not control or influence the population growth within the species. Instead, it is preferred that the species are allowed to thrive based on natural selection.

We addressed both exploration and exploitation aspects in the algorithm as a good ratio between them is key in a heuristic search [19]. Exploration in GASP is achieved during the population initiation where individuals are randomly generated and then speciated into their respective subpopulations. These individuals then evolve to exploit the solution space within their niche. To avoid premature loss of diversity to genetic stagnation, inter-species crossover and mutation are implemented that may lead to spawning of new species or revival of extinct species. Exploitation, on the other hand, is achieved by intra-species crossover and subjecting the new individuals to local competition. A “kill” percentage is implemented as a means of cleansing or replacing weaker individuals with healthier ones. This is to ensure convergence is attained. In GASP, we use two different crossover techniques, one is intra-species crossover, and the other is inter-species crossover. The GASP procedure is shown in Fig. 1.

<i>GASP Algorithm</i>	
1	Begin
2	pop=Rand_pop(); % randomly initialization the population
3	MAXGEN=100; % set maximum generation to 100
4	gen=0; % set generation count to 0
5	fitness=0; % set fitness to 0
6	Max_fit=0;
7	TARGET=Some_number;
8	While (fitness < TARGET) && (gen < MAXGEN)
9	gen=gen+1;
10	fitness=Calc_gen_fitness();
11	Rank_fitness(pop);
12	Replace(pop); % replace bottom 5% with top 5%
13	subpop=Sort(pop); % sort individuals into subpop
14	SUBPOP_SIZE=size(subpop);
15	For i=1:SUBPOP_SIZE
16	SUS_select(subpop); % select solutions
17	intra_xo(subpop); % intra-species crossover
18	inter_xo(); % inter-species crossover
19	Rand_mutate(); % randomly enable/disable links
20	pop=New_pop(); % replace with new individuals
21	End For
22	genome=Genome_map(); % map individual genome
23	Diversity(genome); % calculate population diversity
24	Max_fit=Max_fitness(pop); % get best solution
25	Plot(Max_fit); plot fittest individual trend
26	Plot(subpop); plot subpopulation distribution
27	Plot(Diversity); plot diversity trend
28	End while
29	End

Fig. 1. GASP Procedure

A. Intra-Species Crossover

Intra-species crossover or in-breeding, is a form of phenotype crossover. In intra-species crossover, we apply a single-point crossover at the mid-point of hidden nodes in the NN structure. At this mid-point, the parents’ genetic material are crossover and recombined to produce the new offspring. Intra-species crossover will produce offspring of the same structure retaining its species’ compatibility. This technique safeguards structural innovation and provide time for potential individuals to breed and thrive within the species. Competition for survival is contained within the species to prevent global threat of any one species from taking over the entire population too quickly. We subject 90% of the population to intra-species crossover from which the offspring is reproduced to replace its parents.

B. Inter-Species Crossover

Inter-species crossover or cross-breeding, is a form of genotype crossover whereby we apply a single-point crossover at the mid-point of the genotype between two parents of different species. The resulting child is not guaranteed to be of the same species as the parents. That is, the crossover and recombination at the mid-point of the genotype may produces offspring of outside of the parents’ species. This allows for new search spaces to be explored during the evolutionary process. 1% of the population is subjected to inter-species crossover as the explorative nature of the algorithm is largely undertaken

by a random population initialization at the beginning of the evolutionary process. A small percentage of inter-species crossover ensures that the solutions converge and not diverge causing imbalance to the heuristic search process.

C. Structural and Weights Mutation

In GASP, we introduced a mutation operation to provide a means to restore lost genetic material and prevent possible solutions stagnation. We set a 1.5% probability for an individual's weight to undergo mutation to a random value. We also set a 1% probability for an individual's disabled connection to be re-enabled. These mutation parameters are to potentially change the NN architecture complementing the crossover operations to avoid the solution from being trapped in a local minima.

IV. NN DESIGN USING THE GASP

A. Multi-layer Perceptron NN

There are many variant of neural network depending on the network structure, the connections and the activation function. Feedforward neural network (FFNN) is the most common type of neural network. In FFNN,

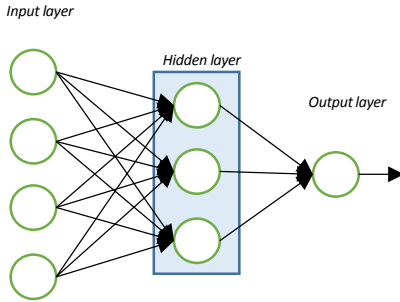


Figure 2. Feedforward MLP NN

information flows only in one direction. One example of a feedforward NN is the multi-layer perceptron (MLP) shown in Fig. 2. The feedforward MLP NN in Fig. 2 consists of 3 layers – input, output and a single hidden layer. Nodes in the input and output layer are interconnected via nodes in the hidden layer. Each connection is associated with a certain weight. The hidden and output layer could have a bias option activated with a constant unit of '1'. The hidden nodes contains a non-linear activation function. Whereas the output nodes usually contains a linear activation function as the output layer computes and sum the results of the process. During the modeling process, NN attempts to map the input variables to the output by constantly adjusting the weights and structure through a learning algorithm guided by an objective function until the desired NN architecture is generated. The objective function is usually in the form of an error function to be minimized. The relationship between the output neuron y to the inputs is govern by,

$$y = f(w^T x + b) \quad (9)$$

where x represents the inputs, w a set of weights, b a bias term of unity value and f is the sigmoid activation function of the hidden nodes,

$$f(x) = \frac{1}{1+e^{-x}} \quad (10)$$

or the linear activation function of the output node,

$$f(x) = (x) \quad (11)$$

The single hidden layer NN is chosen in our example for it is known from the Universal Approximation Theorem that a single hidden layer NN can approximate any arbitrary continuous function [16].

One of the challenges when evolving the structure of an NN is that the connection weights of a new innovation might need a few generations to be optimized. However during the process of crossover of individual with different topology, the child might not have the same topology causing a loss of structural innovation. Our GASP algorithm address this problem by implementing a novel speciation approach customized for NN optimization. In the algorithm, all individuals go through a phenotypic classification process which divides the population into smaller sub-population (species) according to their structural similarity. The individual will then compete within their own species for survival first, before they are exposed to global competition. Through this approach, structural innovation is protected as these structures would be isolated within their own species thereby giving prospective individuals a chance to be optimized.

B. Chromosome, Genotype and Phenotype

In natural genetic, information is carried in discrete elements called genes. The different forms of a gene, called allele are required to synthesize about the individual's physical traits. Specific genes are located in set positions on the chromosomes. The position of these specific genes is known as its locus. The genotype of an individual are its alleles at a locus. The genotype are visually expressed through the individual's phenotype. In another words, phenotypes are the individual's physical traits mapped to its genotype.

To apply GA to solving NN optimization problem, an encoding scheme is devised to translate the possible candidate solutions in the chromosomes. The chromosomes can be of different length to cater to diverse NN architecture. Once designed they are then used as building blocks to construct the genotype to phenotype mapping. The genotype is the coding space and the phenotype is the NN structure search space. GA goal is to find the fittest individual in the genotype space from which the phenotype can be derived.

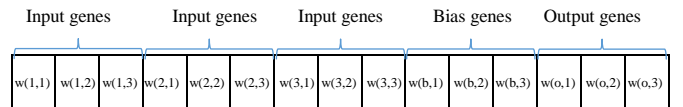


Fig. 3. Chromosome

The chromosomes in Fig. 3 illustrates how the MLP NN in Fig. 2 is encoded. In the chromosome encoding, the input gene $w(i_m, h_n)$ denotes the *input weight* in w for the *connection* from *input layer* i node m to *hidden layer* h node n . The output gene $w(o_l, h_n)$ denotes the *output weight* in w for the *connection* from *output layer* o node l to *hidden layer* h node n . The parameters m , n and l

represent the size of input, hidden and output layers respectively. An *input weight* of '0', that is $w(i_m, h_n) = 0$, represents a disabled connection from input node m to hidden node n . An option for bias to, for instance, the hidden layer can be activated by adding the bias gene $w(b_{m+1}, h_n)$ to the chromosome. Details of the encoding scheme and its design are discussed in our earlier works in [17][18].

Using the chromosome designed in Fig. 3 as building blocks, we construct a chromosome matrix that could easily map its genotype-phenotype correlation for any individual. An example of a chromosome matrix, also known as the genotype, is shown in Fig. 4a. Its corresponding phenotype is shown in Fig. 4b. The genotype is capable of fully embodying the structure of the NN (weights and connections). The genotype is represented as a connection matrix in Fig. 4c, for ease of computation, code handling and NN structure analysis. Fig. 4c shows the connection matrix, `nn_ind`. The input

	Hidden node 1	Hidden node 2	Hidden node 3
Input node 1	0	-0.348	0
Input node 2	0	0	0.492
Input node 3	0.628	0	0.914
Bias Input node 4	-0.583	-0.569	0.239
Output node	0.023	-0.345	0.295

weights at matrix positions; `nn_ind(1,1)`, `nn(1,3)`, `nn_ind(2,1)` `nn_ind(2,2)` and `nn_ind(3,2)` have '0' values.

Fig. 4a. Chromosome Matrix for an NN Individual

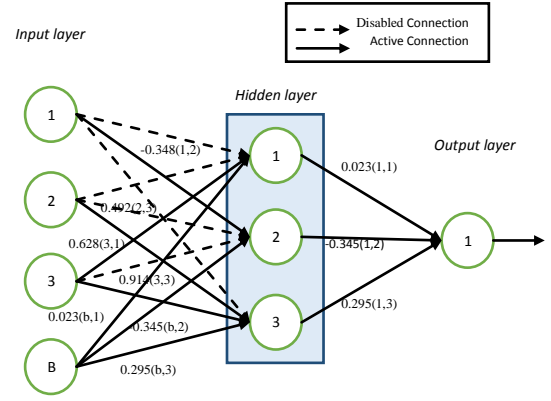
This means that the respective connections from the inputs to the hidden nodes are disabled with the rest of the connections enabled (see Fig. 4b).

The flexibility of the proposed encoding scheme allows the MLP NN structure to be represented in a simple and efficient chromosome matrix, which in turn improves the GASP computational speed.

C. Diversity Measure

In evolutionary computing, the need to measure the diversity of the solution sets stems from the desire to avoid premature convergence of the solutions. However, it is worth noting that high diversity does not necessarily mean the diverse population was obtained by a good balance between exploration and exploitation [19] as a diverse population can be obtained by mere exploration. Also, diversity does not translate to fitness. A diverse population means the individuals are just different. Nonetheless, these genetic differences should be exploited to produce solutions closer to the global optima.

The diversity of individuals or population can either be measured in the genotype or phenotype space as discussed in Section II, part D. As there exists no single measure that fits all the problem definitions [19], a diversity measure designed specific to the problem it aims to solve would be highly effective. In our diversity measure design, we borrow the idea from [20], and adapted for the NN optimization problem.



$$\text{nn_ind} = \begin{bmatrix} 0 & -0.348 & 0 \\ 0 & 0 & 0.492 \\ 0.628 & 0 & 0.914 \\ -0.583 & -0.569 & 0.239 \\ 0.023 & -0.345 & 0.295 \end{bmatrix}$$

Fig. 4c. Connection Matrix

The diversity measure suggested in [20] comes from the concept of linguistic complexity. For a string s defined on a given alphabet A , its linguistic complexity is defined as the ratio of the number of substrings of s , to the maximum of substrings that can be obtained from a string of the same length on the same alphabet. To illustrate, we have the *linguistic complexity*, LC for string s generalized to a population as follows:

$$LC(P) = LC(\{i_1, i_2, \dots, i_n\}) = \frac{|S_{\{i_1, i_2, \dots, i_n\}}|}{\max_{P'_A \sim P} |S_{\{i'_1, i'_2, \dots, i'_n\}}|} \quad (12)$$

where $\max_{P'_A \sim P} |S_{\{i'_1, i'_2, \dots, i'_n\}}|$ is the maximum number of substrings that can be obtained with a population P'_A built on the same alphabet A of P , and having the same number of individuals and with the same length. $LC(P)$ measures the potential motif existence contributed by the kind and number of its individuals.

This gives an idea how effective is the solution exploration being carried out in the genome space relative to what can be done with the same number of individuals, same genome length and on the same alphabet. To draw parallel from the concept of LC , we denote the population P as,

$$P = \{i_1, i_2, \dots, i_n\} \quad (13)$$

where i_j is an individual from population P , whose genome string is S_{i_j} constituting of genomes that appear in the string $\{S_{i_1}, S_{i_2}, \dots, S_{i_n}\}$. The potential motifs is denoted as,

$$S_{\{i_1, i_2, \dots, i_n\}} \quad (14)$$

and its cardinality as,

$$|S_{\{i_1, i_2, \dots, i_n\}}| \quad (15)$$

where $S_{\{i_1, i_2, \dots, i_n\}}$ is the total number of substrings that appears only once in the genome of multiple individuals within the population genome, that is,

$$S_{\{i_1, i_2, \dots, i_n\}} = \bigcup_{j=1}^n S_{i_j} \quad (16)$$

We then define the diversity measure $D(P)$ for a population P in Eq. (13) as follows,

$$D(P) = D(\{i_1, i_2, \dots, i_n\}) = n \frac{|S_{\{i_1, i_2, \dots, i_n\}}|}{\sum_{j=1}^n |S_{i_j}|} \quad (17)$$

where n is the number of individuals in a population, $|S_{\{i_1, i_2, \dots, i_n\}}|$ is the total number of substrings and $|S_{i_j}|$ is the number of substrings in the genome of the individuals considered separately.

Suppose a population has n individuals, and each individual i_j has the same genome s and consequently, the same substrings S , its density measure D , can be shown by applying Eq. 17 as,

$$D(\{i_1, i_2, \dots, i_3\}) = n \frac{|S|}{\sum_{j=1}^n |S|} = 1$$

A diversity measure of 1 is interpreted as all individuals in the population having identical genome, or no diversity. Analogously, if all the individuals' genome in the population are different from each other, we have $D(P) = n$. Therefore, we all always have,

$$1 \leq D(P) \leq n \quad (18)$$

where n is the size of the population.

To utilize the diversity measure in Eq. (17) for our NN optimization problem, a method is devised to translate the individual's genotype or connection matrix to its genome string. To show how the method works, we use the NN example given in Fig. 4c for illustration. Fig. 4c shows the connection matrix, nn_ind. This matrix encapsulates the phenotype meaning of an individual NN shown in Fig. 4b. We implemented a code to convert the non-zero elements in the connection matrix to '1' and leave the rest of the elements unchanged (see Fig. 5). We then locate the matrix positions or the indices of the elements '1'. These indices form the genomes which then constitutes the individual's genome string. The derived genome strings from all individuals within the population are used to calculate the population diversity.

For instance in Fig. 5, the returned indices forming the genome string for this individual i_1 , is $S_{i_1} = \{3, 4, 5, 6, 9, 10, 12, 13, 14, 15\}$.

$$\begin{bmatrix} 0 & -0.348 & 0 \\ 0 & 0 & 0.492 \\ 0.628 & 0 & 0.914 \\ -0.583 & -0.569 & 0.239 \\ 0.023 & -0.345 & 0.295 \end{bmatrix} \Rightarrow \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Fig. 5. Connection Matrix to Genome Conversion

To demonstrate our diversity measure approach, we illustrate how the genome strings are formed using 2 connection matrices as examples; nn_ind1 and nn_ind2 which are 5x3 matrix and 5x4 matrix respectively (See Fig. 6).

$$\begin{aligned} \text{nn_ind1} &= \begin{bmatrix} 0 & -0.348 & 0 \\ 0 & 0 & 0.492 \\ 0.628 & 0 & 0.914 \\ -0.583 & -0.569 & 0.239 \\ 0.023 & -0.345 & 0.295 \end{bmatrix} & \text{nn_ind2} &= \begin{bmatrix} 0.865 & 0 & 0 & -0.521 \\ 0 & -0.649 & 0.211 & 0 \\ 0.170 & 0 & 0.914 & -0.688 \\ 0.873 & -0.109 & -0.311 & 0.129 \\ 0.723 & 0.052 & 0.295 & 0.566 \end{bmatrix} \\ \downarrow & & \downarrow & \\ \text{nn_ind1_g} &= \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} & \text{nn_ind2_g} &= \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix} \\ \downarrow & & \downarrow & \\ S_{i_1} &= \{3, 4, 5, 6, 9, 10, 12, 13, 14, 15\}, & S_{i_2} &= \{1, 3, 4, 5, 7, 9, 10, 12, 13, 14, 15, 16, 18, 19, 20\} \end{aligned}$$

Fig. 6. Genome String Formation

The population diversity $D(\{i_1, i_2\})$ given the example in Fig. 6 with two individuals, i_1, i_2 whose genome strings S_{i_1} and S_{i_2} , and total number of substrings $S_{\{i_1, i_2\}}$ are calculated as follows,

$$S_{i_1} = \{3, 4, 5, 6, 9, 10, 12, 13, 14, 15\}, |S_{i_1}| = 10$$

$$S_{i_2} = \{1, 3, 4, 5, 7, 9, 10, 12, 13, 14, 15, 16, 18, 19, 20\}, |S_{i_2}| = 15$$

$$S_{\{i_1, i_2\}} = \{1, 3, 4, 5, 6, 7, 9, 10, 12, 13, 14, 15, 16, 18, 19, 20\}, |S_{\{i_1, i_2\}}| = 16$$

$$D(\{i_1, i_2\}) = 2 \frac{|S_{\{i_1, i_2\}}|}{|S_{i_1}| + |S_{i_2}|} = 2 \frac{16}{10 + 15} = 1.28$$

Now, suppose both individuals i_1 and i_2 , in a new population, are identical where their genome string $S_{i_1} = S_{i_2} = \{3, 4, 5, 6, 9, 10, 12, 13, 14, 15\}$, then the diversity for this new population, $D_{new}(\{i_1, i_2\})$ is,

$$D_{new}(\{i_1, i_2\}) = 2 \frac{|S_{\{i_1, i_2\}}|}{|S_{i_1}| + |S_{i_2}|} = 2 \frac{10}{10 + 10} = 1$$

In another words, $D_{new}(\{i_1, i_2\})$, is a homogeneous population of clones.

V. RESULTS AND DISCUSSION

In this work, we use the building energy dataset obtained from MatLab as benchmark dataset for our experiments. The dataset consists of 14 attributes or inputs for over 4000 different households. Inputs 1 to 10 represents coded day of week and time of day. Inputs 11 to 14 represents temperature, humidity, solar strength and wind respectively. The output is energy usage, to be estimated from the inputs.

The dimension for the connection matrix is chosen to be 16x14 specific to this optimization problems. That is, it is able to accommodate any NN structure with 14 inputs and a bias option (total 15 inputs), and 1 output. The NN is "free" to evolve to any number of hidden nodes up to a maximum of 14. As empirically-derived rule-of-thumb suggests the number of hidden nodes be between that of the input and output [21].

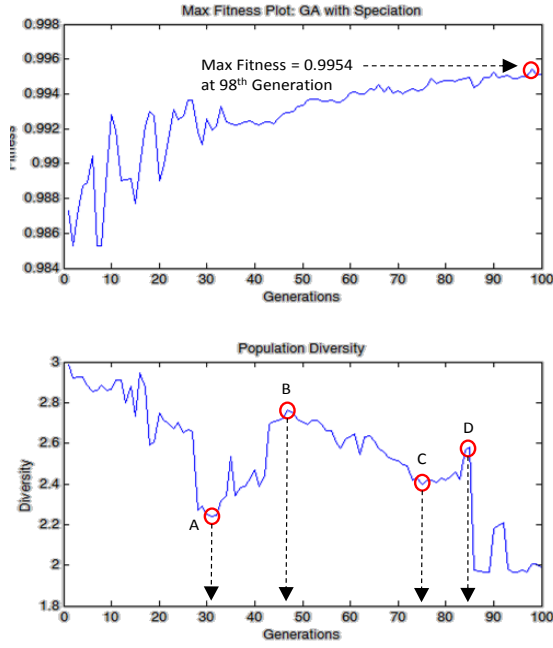


Figure 7. Max Fitness and Diversity Plot

We run the GASP algorithm for 100 generations using 2000 data points from the energy building dataset. 70% is used for training and 30% for testing. Fig. 7 shows the fitness and diversity plot. The fittest NN has a fitness of 0.9954 (maximum fitness is 1) which is found at the 98th generation. During the evolution process, the population diversity measure started at slightly above 3.0. It then declined to point A, at the 31st generation before it rose to point B at the 46th generation. It started descending from point B to C at the 75th generation where population diverged over the next 10 generations to point D. This evolution process has an average diversity of 2.5227. Its genotype is shown in Fig. 8. The optimized NN has 5 hidden nodes and several disabled links indicated by a weight value of '0'. The connection matrix for this NN is highlighted within the bolded square of the genotype in Fig. 8.

We conducted two separate experiments for GASP and SGA to compare the solutions of these two algorithms in relations to the diversity measure, convergence rate and the complexity of the optimized NN. Table 1 shows the experimental results conducted for 20 runs. Both the algorithms has generated good solutions with a mean fitness of 0.9961. However, GASP has a slightly better convergence rate where the best individual was found at an average of 91 generations compared to SGA's 94. The individual with the optimized NN structure found is less complex for GASP with an average structure consisting of 5.55 hidden nodes compared to SGA's 6.70. In terms of population diversity, GASP has a slightly more diverse population measured at 2.4890 to SGA's 2.1419.

However, from our experiment, it is not conclusive that this higher diversity has translated to fitter solutions (see Fig. 9 and 10). Nonetheless, it is important to recognize that diversity improves exploration, as in our case during the earlier evolution process, thereafter exploitation takes over to narrow the search at the expense of lost diversity in order for the solutions to converge.

Input/ Output Nodes	Hidden Node 1	Hidden Node 2	Hidden Node 3	Hidden Node 4	Hidden Node 5
Input 1	0.8566	-0.3664	-0.0709	0.4096	0.5871
Input 2	0.7117	0.0000	0.0000	0.0000	0.0000
Input 3	0.0836	0.2513	0.7073	0.0000	-0.6422
Input 4	-1.0000	0.0000	0.6145	-0.2188	0.0397
Input 5	0.0000	0.3458	0.6102	-1.0000	0.0000
Input 6	-0.3450	-0.3514	0.9497	0.1742	0.1901
Input 7	-0.4915	1.0000	0.0000	0.6388	0.0000
Input 8	0.3986	0.0000	0.0000	0.2276	-0.5906
Input 9	0.0000	-0.2214	0.7887	0.5477	0.0000
Input 10	0.9018	0.6282	0.8812	-1.0000	0.0000
Input 11	0.0000	0.0938	0.5128	-0.7773	0.6997
Input 12	0.4535	-0.5722	0.6243	0.3277	0.0000
Input 13	0.0000	0.5038	0.0000	0.0000	0.5273
Input 14	-1.0000	0.7632	0.3966	-1.0000	0.0842
Bias 1	0.8809	0.8072	-0.2944	0.9287	0.3467
Output 1	-0.3541	0.1782	0.2602	-0.0134	0.6081

Fig. 8. Genotype of Best Individual

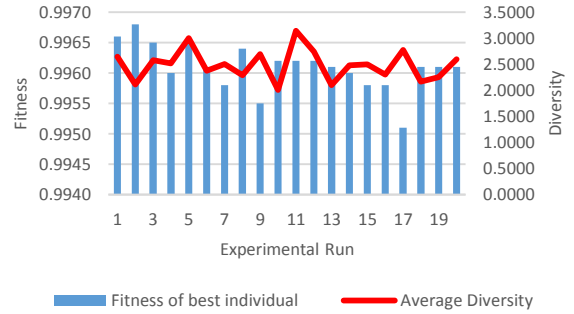


Fig. 9. Fitness and Diversity Chart for GASP

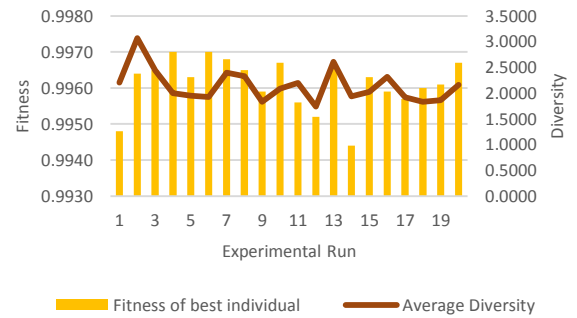


Fig. 10. Fitness and Diversity Chart for SGA

VI. CONCLUSION AND FUTURE WORK

Having the ability to make predictions of a system's future outputs has advantages in numerous modeling and forecasting problems. The GASP optimized NN is seen to lead to better management and resource planning for building energy consumption. In particular, our results show that speciation is a favorable option for solving NN optimization problems with (or without a potentially)

multimodal search space at reduced NN complexity. Subpopulating into species allows these “niches” to search their attractor regions to find the optimal solutions in their own niche. Competition largely occurs within the species which isolate for structural innovation.

We have also presented a novel approach to map individual’s phenotype to its unique genome string. The distance between individual strings and population can be derived from the single string and substrings counts. For future work on diversity measure, we shall explore the inclusion of NN’s weights and layers in the mapping the NN phenotype to its genome string. We are also to further the NN speciation scheme to support multi-layers NN speciation to support deep learning applications.

TABLE 1 EXPERIMENT RESULTS FOR GASP AND GSA

Run	Fitness of best individual		Best Individual Found (Gen)		Average Diversity		Structure of Best Individual(# of HD)	
	GASP	SGA	GASP	SGA	GASP	SGA	GASP	SGA
1	0.9966	0.9948	76	91	2.6479	2.2074	4	5
2	0.9968	0.9964	93	100	2.1132	3.0674	9	4
3	0.9965	0.9965	91	100	2.5755	2.4422	5	6
4	0.9960	0.9970	100	99	2.5188	2.0019	5	8
5	0.9965	0.9963	65	89	3.0034	1.9475	4	7
6	0.9961	0.9970	81	99	2.3767	1.9259	6	7
7	0.9958	0.9968	99	97	2.5054	2.3964	5	5
8	0.9964	0.9965	79	89	2.2898	2.3271	6	5
9	0.9955	0.9959	96	65	2.6927	1.8340	4	8
10	0.9962	0.9967	100	99	2.0065	2.0845	9	7
11	0.9962	0.9956	89	99	3.1430	2.1988	5	6
12	0.9962	0.9952	99	93	2.7448	1.7409	4	10
13	0.9961	0.9966	100	93	2.1009	2.6106	8	5
14	0.9960	0.9944	91	75	2.4771	1.9363	6	9
15	0.9958	0.9963	99	95	2.4984	2.0255	5	8
16	0.9958	0.9959	100	100	2.3004	2.3194	6	5
17	0.9951	0.9957	100	99	2.7720	1.9211	4	7
18	0.9961	0.9960	84	100	2.1683	1.8314	6	7
19	0.9961	0.9961	98	99	2.2511	1.8611	5	9
20	0.9961	0.9967	87	93	2.5937	2.1587	5	6
Mean	0.9961	0.9961	91	94	2.4890	2.1419	5.55	6.70
STDEV	0.0004	0.0007	10	9	0.2988	0.3202	1.54	1.63

REFERENCES

- [1] F. Vafaee, G. Turan, P.C. Nelson and T.Y. Berger-Wolf, "Balancing the exploration and exploitation in an adaptive diversity guided genetic algorithm," IEEE Congress on Evolutionary Computation (CEC), 2014, pp. 2570 - 2577.
- [2] D. Gupta and S. Ghafir, "An overview of methods maintaining diversity in genetic algorithm," International Journal of Emerging Technology and Advanced Engineering, 2012, vol. 2(5), pp. 56-60.
- [3] B. Sareni and L. Krahenbuhl, "Fitness sharing and niching methods revisited," IEEE Transactions on Evolutionary Computation, 1998, Vol. 2(3), pp. 97-106.
- [4] J. H. Holland, "Adaptation in natural and artificial systems," Ann Arbor, MI: Univ. of Michigan Press, 1975.
- [5] D. E. Goldberg and J. Richardson, "Genetic algorithms with sharing for multimodal function optimization," in Proc. 2nd Int. Conf. Genetic Algorithms, J. J. Grefenstette, Ed. Hillsdale, NJ: Lawrence Erlbaum, 1987, pp. 41-49.
- [6] R. W. Morrison and K. A. DeJong, "Measurement of population diversity," in Proc. 5th European Conference on Artificial Evolution 2001, pp. 31-41.
- [7] K. A. DeJong, "An analysis of the behavior of a class of genetic adaptive systems," PhD. dissertation, Univ of Michigan, Ann Arbor, 1975.

- [8] S. Mahfoud, "Crowding and preselection revisited," in Proc. PPSN, 1992, pp.27-36.
- [9] O. Mengshoel and D. Goldberg, "Probabilistic crowding: Deterministic crowding with probabilistic replacement," in Proc. CECCO, 1999, pp. 409-416.
- [10] P. A. Diaz-Gomez and D. F. Hougen, "Initial population for genetic algorithms: a metric approach," in Proc. International Conference on Genetic and Evolutionary Methods, 2007.
- [11] T. Back, "Evolution Algorithm in Theory and Practice. Oxford Univ. Press, 1996.
- [12] L. Li and K. Tang, "History-based topological speciation for multimodal optimization," IEEE Transactions on Evolutionary Computation, 2014, vol. 19(1).
- [13] J. Li, X. Li and A. Wood, "Species based evolutionary algorithms for multimodal optimization: a brief review," in Proc. CEC, 2010, pp. 1-8.
- [14] X. Li, "Efficient differential evolution using speciation for multimodal function optimization," in Proc. GECCO, 2005, pp. 873-880.
- [15] A. D. Cioppa, C. De Stefano and A. Marcelli, "Where are the niches? Dynamic fitness sharing," IEEE Trans. Evolution Computing, 2007, vol. 11(4), pp. 453-465.
- [16] S. Haykin, Neural Networks: A Comprehensive Foundation, 2nd Edition, Prentice-Hall International, Upper Saddle River, N.J, 1999.
- [17] Y.W. Foo, C. Goh, H.C. Lim and Y. Li, "Evolutionary Neural Network Modeling for Energy Prediction of Cloud Data Centers," International Symposium on Grids and Clouds, 2015.
- [18] Y. W. Foo, C. Goh, H.C. Lim, Z-H. Zhan and Y. Li, "Evolutionary Neural Network Based Energy Consumption Forecast for Cloud Computing," International Conference on Cloud Computing Research and Innovation, 2015.
- [19] M. Crepinsek, S. Liu and M. Mernik, Exploration and exploitation in evolutionary algorithms: a survey," ACM Computing Surveys (CSUR), 2013, vol. 45(3), pp. 35.
- [20] C. Mattiussi, M. Waibel and D. Floreano, "Measures of diversity for populations and distances between individuals with highly reorganizable genomes," Evolutionary Computation Journal, 2004, vol. 12(4) pp. 495-515.
- [21] J. Heaton, Introduction to Neural Networks for Java, 2nd Edition, Heaton Research, 2008.