Abstract

The problem considered is the estimation of the evolutionary history of a collection of organisms in terms of a phylogenetic tree using genetic algorithm. This paper presents a new chromosome representation scheme for individuals in the population. A new stack based fitness evaluation method has been proposed to handle the populations generated in genetic algorithms. The problem of phylogenetic tree reconstruction is a hard combinatorial optimization problem for which different Evolutionary Algorithmic approaches are proposed and evaluated. It is shown that Genetic Algorithm for phylogenetic tree reconstruction by maximum parsimony method with the proposed stack based fitness evaluation scheme and with appropriate individual representation scheme performs well with reasonable storage space and increased computational speed.

Keywords Genetic Algorithm; phylogenetic tree; maximum parsimony; fitness evaluation; position based crossover; chromosome representation; mutation; selection; population.

1. Introduction

The inference of phylogenetic tree is one of the most important and challenging tasks in Biology. Such trees are used to represent the evolutionary history of a collection of organisms or taxa from their molecular sequence data. An accurate estimation of this evolutionary history is a very useful tool in many areas of Biology, such as multiple sequence alignment, molecular epidemiological studies, drug designing, etc.

Unfortunately, the problem of phylogenetic tree reconstruction constitutes a very hard combinatorial optimization problem for most optimality criteria. Exact techniques such as branch-and-bound can be used, but would be computationally unaffordable for even moderate, say, 30-40 taxa. Hence the use of genetic algorithm seems appropriate.

This paper is concerned about the utilization of genetic algorithm [1] for tackling the phylogeny problem. As an initial attempt to apply genetic algorithms to phylogenetic inference, maximum parsimony method has been chosen, since it is not computationally demanding when compared to maximum likelihood. Section 2 of the paper details about phylogenetic trees and their reconstruction methodologies in general. Section-3 briefs about Genetic Algorithm. The proposed genetic algorithm, along with tree representation, stack based fitness evaluation, other operators chosen is explained in section 4 and Section-5 concludes the paper.

2. Phylogenetic Tree Reconstruction

A phylogenetic tree[5] is a sketch portraying the course of evolution. Evolutionary changes may be modeled as a process by a series of bifurcations: Species A gives rise to a new species B, and both move forward in time to produce another species say C and so on. Phylogenetic trees may be rooted or unrooted. Generally these trees are constructed from molecular data obtained as nucleotide or protein sequences. Nucleotide or DNA sequences, are comprised of the characters A,C,G and T. Advances in the availability of molecular data have given rise to several techniques for analyzing the ancestral relationships between sequences. The methods can be broadly categorized as distance based and character based [6]. Distance based methods depends on the pair wise alignment and the distance between sequences, ignoring the actual characters present in each site of the sequences. They are easy to manipulate and can be used as a startup for other advanced character based methods. Character based methods are more accurate, at the same time being more complex. Maximum parsimony falls under the character based techniques.
Maximum-parsimony aims at finding the tree that explains the observed pattern of nucleotides with the smallest number of mutations possible [7]. This is done by placing substitutions on possible tree topologies so as to minimize the total number of substitutions required to explain the nucleotide at each branch tip. The most parsimonious reconstruction, i.e. the one requiring the fewest substitutions, is chosen for each site. The total number of evolutionary changes on a tree is the sum of the number of changes for each site, and the tree topology that has the fewest changes is chosen as the most parsimonious tree.

Maximum-likelihood [8][11] is the approach that is generally considered to make the most efficient use of the data and to provide the most accurate estimates of a phylogenetic tree. It is the most computationally demanding technique for reconstructing phylogenetic trees even for small data sets and number of sequences greater than 6 or 7. The basic idea of the likelihood approach is to compute the probability of the observed data assuming it has evolved under a particular evolutionary tree and a given probabilistic model of nucleotide or amino acid substitution. The maximum-likelihood (ML) tree is the tree with the highest probability of explaining the data.

2.1 Informative Sites

All sites in a set of sequences does not matter in the tree reconstruction. Only the informative sites are considered for tree reconstruction. An informative site is one, which contains at least two different nucleotides that are shared by at least two taxa each. To find the true tree, we can discard all sites that aren’t informative. By using only informative sites to construct trees, data processing requirements are greatly reduced and a substantial speed up of computer searches is obtained.

Another problem in phylogenetic tree inference is the number of trees or topologies generated. The number of unrooted trees grows very quickly with n, where n is the number of Operational Taxonomic Unit (OTU) or taxa. Given n taxa, the number of possible different bifurcating unrooted trees is given by the formula shown below and is presented in table-1 [9]:

\[
\frac{(2n - 5)!}{2^{n-3}(n - 3)!}
\]

<table>
<thead>
<tr>
<th>Sequences</th>
<th>Unrooted Trees</th>
<th>Rooted Trees</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>15</td>
</tr>
<tr>
<td>5</td>
<td>15</td>
<td>105</td>
</tr>
<tr>
<td>6</td>
<td>105</td>
<td>945</td>
</tr>
<tr>
<td>7</td>
<td>945</td>
<td>10395</td>
</tr>
<tr>
<td>8</td>
<td>10395</td>
<td>135135</td>
</tr>
<tr>
<td>9</td>
<td>135135</td>
<td>2027025</td>
</tr>
<tr>
<td>10</td>
<td>2027025</td>
<td>34459425</td>
</tr>
<tr>
<td>20</td>
<td>2.22E020</td>
<td>8.20E21</td>
</tr>
<tr>
<td>30</td>
<td>8.69E036</td>
<td>4.95E038</td>
</tr>
</tbody>
</table>

Table 1. Number of possible Tree Topologies

3. Genetic Algorithm

Genetic Algorithm [1][11] is an adaptive heuristic search algorithm based on the evolutionary ideas of natural selection and genetics. As such they represent an intelligent exploitation of a random search used to solve optimization problems. Although randomized, GA is by no means random, instead they exploit historical information to direct the search into the region of better performance within the search space. GAs are designed to simulate processes in natural systems necessary for evolution; specially those that follow the principles first laid down by Charles Darwin of “survival of the fittest”. Since in nature, the competition among individuals for scantly resources, results in the fittest individuals dominating over the weaker ones.

In the traditional genetic algorithm, the representation used is a fixed-length bit string. Each position in the string is assumed to represent a particular feature of an individual, and the value stored in that position
represents how that feature is expressed in the solution. Usually, the string is “evaluated as a collection of structural features of a solution that have little or no interactions”. The analogy may be drawn directly to genes in biological organisms.

The general structure of genetic algorithm[1] is presented below.

1. Generate random population of n chromosomes (suitable solutions for the problem).
2. Evaluate the fitness \( f(x) \) of each chromosome \( x \) in the population.
3. Create a new population by repeating the following steps until the new population is complete.
4. Select two parent or chromosomes from a population according to their fitness (the better fitness, the bigger chance to be selected).
5. With a crossover probability, crossover the parents to form a new offspring. If no crossover was performed, offspring is an exact copy of parents.
6. With a mutation probability mutate new offspring at each position in chromosome
7. Place new offspring in a new population.
8. Use newly generated population for a further run of algorithm.
9. If the end condition is satisfied, stop, and return the best solution in current population.
10. Go to step 2.

Step 1 involves a suitable representation of the initial population. Step 2 is the evaluation phase which involves quantifying the fitness of each individual, i.e. how well each individual attempt at the problem solution. The most common approach to measure fitness in GA is to assign each individual in a population a ‘fitness measure’. The final fitness of a population is a sum or an average of several different fitness cases. The fitness can then be measured against the correct answer to the problem. The are various fitness measures used based on the nature of problem. Some of them are Raw fitness, Standardized fitness, Adjusted fitness and Normalized fitness [2].

After fitness evaluation, suitable selection methods are used to pick up the individual to be mated for producing the offspring. Selection pressure is the probability of the best individual being selected compared to the average probability of selection of all individuals. Depending on the aims and the time available, the selection pressure can be changed to achieve a different final result.

Applying certain genetic operators on the selected parents does the creation of new individuals. The most common genetic operators are reproduction, mutation and recombination. Reproduction and mutation are unary operators. Reproduction simply copies the individual into the new generation, while mutation modifies its argument by randomly changing each gene of it with a certain probability. Recombination takes two individuals and creates new ones by exchanging parts of their gene-chains.

4. Proposed Genetic Algorithm for Phylogenetic Tree Reconstruction

The proposed system is an application of genetic algorithm for phylogenetic inference [11][12]. Hence various issues like chromosome representation, fitness evaluation, selection criteria, operators etc. have to be considered and appropriately designed to suit the problem. Each of the issues listed below are considered and appropriately designed.

- Initial Population Generation
- Fitness Calculation
- Parent Selection
- Operator selection

4.1 Initial Population Generation

This first step in GA is initial population generation. A random number of individuals are generated for a given number of species. A user interface is provided for setting the parameters for initial population. Once the initial parameters are chosen for the Genetic algorithm, the individuals are generated based on the representation and method provided in the next section.
4.1.1 Individual Representation

The individuals are represented in the form of a string of characters called genomes. The genome consists of taxas or so called species. Each position in the genome reflects the taxa. The taxa must be unique in each chromosome. The individual representation for four taxa is given below.

<table>
<thead>
<tr>
<th>Individual</th>
<th>12,34,</th>
</tr>
</thead>
<tbody>
<tr>
<td>Individual2</td>
<td>13,24,</td>
</tr>
<tr>
<td>Individual3</td>
<td>14,23,</td>
</tr>
</tbody>
</table>

For five species, the 5th species is added at five different positions in each individual specified above. Therefore the total population for five species is 5 x 3 = 15 individuals. The new species is added along with a comma, retaining the existing commas. For example, when the 5th species is added to individual 12,34, the five new individuals obtained is shown below.

<table>
<thead>
<tr>
<th>Individual1</th>
<th>15,2,34,</th>
</tr>
</thead>
<tbody>
<tr>
<td>Individual2</td>
<td>125,,34,</td>
</tr>
<tr>
<td>Individual3</td>
<td>12,5,34,</td>
</tr>
<tr>
<td>Individual4</td>
<td>12,35,4,</td>
</tr>
<tr>
<td>Individual5</td>
<td>12,345,,</td>
</tr>
</tbody>
</table>

The general algorithm for initial population generation is given below:

1. Start with the initial topology for four taxa.
2. Construct the topologies for next taxa by adding the taxa to all possible positions in each individual of initial topology.
3. Make the resultant topologies as the initial topology.
4. Repeat Step 2 until the topologies are constructed for the given number of taxa.

Since a string representation is used to represent the individuals in the population, a permutation of the individual species in the string is sufficient to generate the required set of topologies for the initial population.

4.2 Proposed Stack Based Fitness Calculation

The proposed stack based fitness calculation works well with the given chromosome representation. Stacks are used to group the characters. This reduces storage space and increases computational speed. The algorithm for fitness calculation is based on Fitch algorithm [10] and a stack based representation is given below.

1. Push characters into stack till a ‘,’ is encountered.
2. Pop out the characters from the stack
3. Check characters.
4. If they are same, intersect the characters.
   else
      Find the union of characters and increment the parsimony score.
5. Push the result into stack.
6. Repeat the above steps till the end of the chromosome is reached.

Once the fitness of individuals in each generation is evaluated, suitable individuals are selected for mating. Roulette wheel selection methodology [1] is used here. The crossover operator used in the program is Position based crossover [3]. The mutation operator used is Reciprocal Exchange. A copy of the best individual of the population is always passed to the new generation. It is accomplished through Elitist operator [1].
4.3 Results and Statistics

The performance of the genetic algorithm for phylogenetic tree reconstruction was determined using real data sets pertaining to T.Solium and other parasites obtained from the Department of Biotechnology, Pondicherry University, Pondicherry, India. About 10 nucleotide sequences, each of length around 13,000 bp, pertaining to parasites was taken up for determining the evolutionary pattern. The parameters for the algorithm were set using the user interface provided. The program was run for 50 generations with 50 number of individual in each generation. Roulette wheel fitness evaluation was used. 90% of crossover and 10 % of mutation was found to produce good results. The program was run varying the number of sequences and the time taken to infer the phylogenetic tree for varying number of sequences is provided below.

<table>
<thead>
<tr>
<th>Number of Sequences</th>
<th>Time Taken (min)</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>2</td>
</tr>
<tr>
<td>7</td>
<td>10</td>
</tr>
<tr>
<td>8</td>
<td>20</td>
</tr>
<tr>
<td>9</td>
<td>32</td>
</tr>
</tbody>
</table>

Table-2: Time taken for tree reconstruction

5. Conclusion

To explore the efficiency and intricacies of genetic algorithms for solving problems of large solution space, the problem of phylogenetic tree reconstruction from bioinformatics has been taken up. This new approach to maximum parsimony based tree reconstruction will support this endeavor, and promote new insights. In this attempt, a suitable representation for the trees has been tried and a new stack based fitness evaluation strategy has been proposed. The individuals of the population, though they are trees, are encoded as character strings for easy manipulation as well as to suit the nature of genetic algorithm. The proposed stack based fitness evaluation method was found to conveniently handle the fitness calculation of the individuals.

As the problem is NP-complete in nature, the use of GA and the associated fitness evaluation scheme proposed has produced good results for the real data obtained from the Department of Biotechnology in limited time and space. Thus the suitability of GA to solve problems with large data sets and search space has been tried and implemented successfully on Phylogenetic Tree Reconstruction.

As the project is an initial attempt to use GA for a problem of biological domain, a simple method namely maximum parsimony is chosen to reconstruct phylogenetic tree. In future, the same technique could be applied to reconstruct phylogeny using the more accurate and compute intensive method of maximum likelihood. As the sequences are very large, parallel GA can be deployed to improve the computational speed by distributing the initial population to a number of workstations.

6. References