Using directed evolution techniques to solve hard combinatorial problems

Paula Cordero Moreno
Natural Computing Group - Universidad Politécnica de Madrid
28660 Madrid, Spain
e-mail: p.cordero@alumnos.upm.es

Angel Goñi Moreno
Natural Computing Group - Universidad Politécnica de Madrid
28660 Madrid, Spain
e-mail: agmoreno@gcn.upm.es

Juan Castellanos Peñuela
Artificial Intelligence Department - Universidad Politécnica de Madrid
28660 Madrid, Spain
e-mail: jcastellanos@fi.upm.es

ABSTRACT

The study of new computation paradigms as suitable methods for the resolution of hard mathematical problems is the starting point of this proposal. In this work it has been developed a theoretical idea that incorporates mechanisms of directed evolution in-vitro, mutation (site-directed mutagenesis) and recombination (DNA Shuffling) in order to define algorithms and models whose aim is the resolution of the Hamiltonian Path Problem. It is also proposed in the paper a crossover operator for a genetic algorithm whose population of individuals is made up of plasmid vectors.

Keywords

Computing complexity, NP-problems, Directed evolution, Mutagenesis, DNA Shuffling, Genetic algorithm, Crossover operator.

1. INTRODUCTION

Nowadays the computers are able to solve problems by using algorithms that have a polynomial complexity or computational cost at the most. The problems included in the complexity set called NP-hard, are very studied in the past decades because of the difficult resolution of those in conventional computers. Due to that reason, different research areas try to look for possible alternative solutions. One of them is natural computing which gets a strong inspiration in nature and biology for the definition of algorithms oriented to the practical resolution of this kind of problems.

Following this scientific line it appeared in 1970 a genetic algorithm (GA) definition [1][2]. GA's are inspired on the biological evolution process and its genetic-molecular principles. These algorithms evolve a population of individuals by running on them similar procedures to the biological evolution functions. The objective is to solve a hard combinatorial problem. Leonard M. Adleman [5] was the creator of the first implementation of a computer based on DNA operations, this computer solved a NP-problem (HPP) using desoxirribonucleic acid molecules. Computing using a DNA.

All the developments presented in this paper are based on these ideas and they try to propose new methods for the application of biological mechanisms in the design of algorithms for the resolution of highly complex problems.

2. COMPUTING COMPLEXITY AND PARALLEL COMPUTING

Computational complexity theory is a branch of the theory of computation in computer science that investigates the problems related to the resources required to run algorithms, and the inherent difficulty in providing algorithms that are efficient for both general and specific computational problems.

The most studied problems are included in the complexity class *NP* that is the set of decision problems that can be solved by a non-deterministic Turing machine in polynomial time. This class contains many problems that people would like to be able to solve effectively including the Boolean satisfiability problem or the Hamiltonian path problem. In order to solve such problems many people study the possibility of putting into practice genetic algorithms based on proposal from J.Holland[1][2].

2.1. Hamiltonian Path Problem (HPP)

The Hamiltonian path problem and the Hamiltonian cycle problem are problems of determining whether a Hamiltonian path or a Hamiltonian cycle exists in a given graph [5]. A Hamiltonian cycle is a cycle in an undirected graph which visits each vertex exactly once and also returns to the starting vertex. Both problems are NP-complete in the mathematical field of graph theory.

2.2. Simple genetic algorithm (GA)

A GA is a massive parallel mathematical algorithm that turns a set of mathematical objects or individuals into a new and more adapted set. It uses operations modeled according to the Darwinian principle of reproduction and survival of the fittest. There must be defined in a natural and formal way several genetic operations. Between them, it is important to emphasize the recombination process. Each individual is usually identified with a chain of characters of fixed length that corresponds to a chains of chromosomes. It is associated to each chain a certain mathematical function that reflects its aptitude. [2]

They are global search heuristics based on probability. Under a very weak condition based on elitism it can be demonstrated that the algorithm converges in probability to the optimal. In other words, when increasing the number of iterations, the probability of getting the optimal of the population tends to 1.

3. DIRECTED EVOLUTION

Directed evolution describes a set of techniques for the iterative production, evaluation and selection of variants of a biological sequence, usually a protein or nucleic acid.

Numerous protein engineering experiments have demonstrated that changes in protein properties are brought about by the cumulative effects of many small adjustments, many of which are distributed or propagated over significant distances. It is possible to produce new enzymes in recombinant organisms, altering the amino acid sequence and therefore the properties through appropriate modifications at the DNA level.

3.1. Site-directed mutagenesis

Using site-directed mutagenesis [4] the information in the genetic material can be changed. A synthetic DNA fragment is used as a tool for changing one particular code word in the DNA molecule. This reprogrammed DNA molecule can direct the synthesis of a protein with an exchanged amino acid.

One way to do site specific mutagenesis is to start with the gene you want to change in double strand form. The strands of the DNA are separated by heating. The oligo is present, and it will hybridize with its complementary strand. This newly made double strand DNA is like the original DNA double helix, except that one of the bases has been changed, and the base change has been directed by the sequence of the oligo that was used. The process is then repeated, and more mutated DNA is produced. This method is known as "PCR" mutagenesis.

3.2. DNA Shuffling

DNA shuffling [6] is a powerful process for directed evolution, which generates diversity by recombination, combining useful mutations from individual genes. Libraries of chimaeric genes can be generated by random fragmentation of a pool of related genes, followed by reassembly of the fragments in a self-priming polymerase reaction. Template switching causes crossovers in areas of sequence homology.

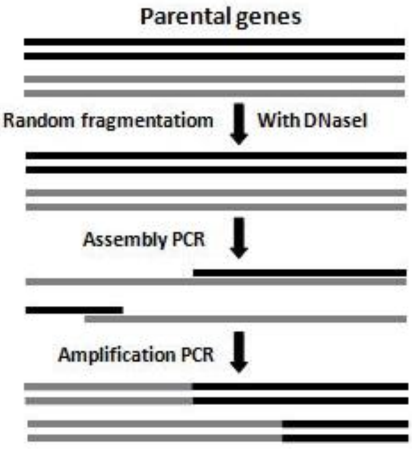

Figure 1. Parental genes are randomly fragmented using DNaseI.

4. COMPUTING APPLICATIONS

4.1. HPP solved by mutagenesis

In this section it will be explained, in general, a possible computational method of one step included in the resolution of a complete algorithm that solves the well-known Hamiltonian Path Problem (NP-Complete Problem). When we talk about directed graphs with a different weight on each edge, the Hamiltonian cycle (Cy) with a smaller cost (Co = $\sum$ edge_weigh $\forall$edges of Cy) is also known as the solution to the Traveling Salesman Problem.

As this investigation advance we part from a codified population of ways using plasmids or vectors, as we will see next. This population must have been created following successive steps of a determined algorithm in such a form that finally we could get all the possible combinations of edges of the graph in order to obtain all the possible solutions of the problem.

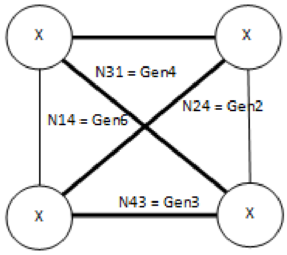

Figure 2.The graph depicts HPP.

Concretely, it will focus the problem on the graph illustrated (figure 2). In the initial set of paths we could find valid, invalid, complete and incomplete solutions. That is why the procedure explained next is so important.

Each problem is represented on a certain graph. Not all the graphs will have a Hamiltonian Path (figure 3B). The purpose of final algorithm to solve HPP is to tell us if there is a solution in our graph or not.

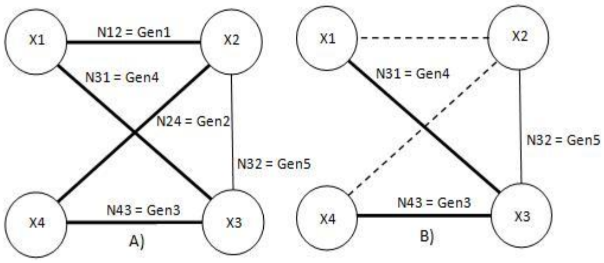

Figure 3. The graph 3A would have a positive solution. A negative solution would be given under graph 3B.

The theoretical development of the presented proposal is based on a specific codification of the existing edges in the graph of the problem using nucleotides sequences which codify concrete genes. These genes will be expressed in fluorescents proteins when transcribed. Each edge of the graph will be codified as shown in figure 4. The fitness field representing the fluorescent gene is surrounded by bond-free sequences which represents half of the vertex.

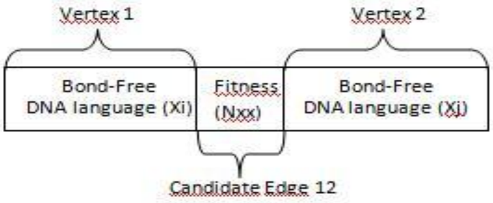

Figure 4. Edge of the graph codified.

Next there is a table which represents the correspondence between the existing edges in the graph (figure 3A) and the genes associated. The genes represented in the explanatory example are normal nucleotide sequences chosen randomly in order to simplify and clarify the method applied. However, there could be taken into account several possibilities to select the family of genes whose expression result to be fluorescent proteins to carry out the experiment. In this case we could design the experiment with proteins of the type GFP (the green fluorescent protein) and work with the variants that GFP produces like the RFP protein (red fluorescent protein) or YFP (yellow fluorescent protein).

An example of edge codification:

EDGE CODIFICATION	FITNESS GENES
N12: **BF-DNA lang. (X1) - AAT-TGG-CGA-TTA-AAC - BF-DNA lang. (X2)**	TTAACCGCTAATTTG AATTGGCGATTAAAC
N24: **BF-DNA lang (X2) - TTA-CCA-TGC-TGA-CCC - BF-DNA lang (X4)**	AATGGTAGGACTGGG TTACCATCCTGACCC
N43: **BF-DNA lang. (X4) - GGT-CAG-CTG-ACG-TCA - BF-DNA lang. (X3)**	CCAGTCGACTGCAGT GGTCAGCTGACGTCA
N35: **BF-DNA lang. (X3) - AGT-CGA-TTC-GAA-GGC – BF-DNA lang. (X5)**	TCAGCTAAGCTTCCG AGTCGATTCGAAGGC
N51: **BF-DNA lang. (X5) - CGT-AGC-TGA-TCGA-TCT – BF-DNA lang. (X1)**	GCATCGACTAGCTAGA CGTAGCTGATCGATCT
N45: **BF-DNA lang. (X4) - GGC-TGA-TCG-TAA-AGT – BF-DNA lang. (X5)**	CCGACTAGCATTTCA GGCTGATCGTAAAGT
N14: **BF-DNA lang. (X1) - CCG-TAG-CTG-ATC-GTC – BF-DNA lang. (X4)**	GGCATCGACTAGCAG CCGTAGCTGATCGTC

Each plasmid contains sequentially the possible vertexes which form the Hamiltonian path we are looking for. As it has been mentioned before, within the set of plasmids there will be incomplete sequences and also nonvalid chains. For that reason the valid solution must have been searched. The proposal presented in this paper is based on this situation. In figure 5 is shown the plasmid of the initial set which contains the solution sequence to our problem in order to facilitate the later understanding of the theoretical development.

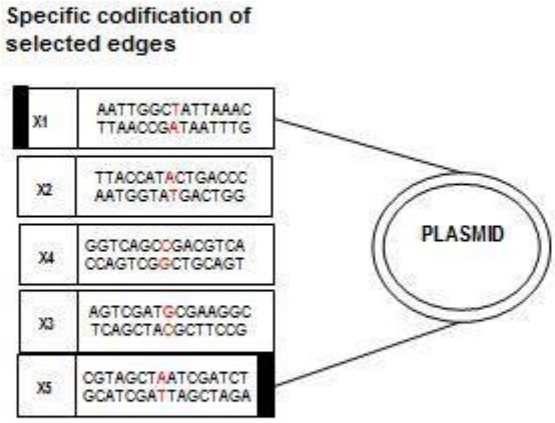

Figure 5. Possible solutions of the problem come expressed into vectors or plasmids

It is necessary to emphasize an important detail in the codification of the plasmid candidates. By observing figure 5 it can be seen that between each pair of vertexes the candidate edges are codified. These edges are codified carefully with a constant length of N-nucleotides and for the case of the existing edges in the initial graph a modification of the central nucleotide of the sequence which corresponds to the fitness gene take place. This modification consists of:

- Central sequence of the original edge N12: AATTGGCGATTAAAC
- Central sequence of the edge codified in the plasmid N12: AATTGGCTATTAAAC
  TTAACCGATAATTTG

It has been already explained the formation of the candidates which could be a solution of the problem. Next it is detailed the rest of the method which detects a correct solution. This technique is based on running n-cycles of mutagenesis in such a way that the solutions that contain the sequence of vertexes corresponding to the Hamiltonian Path will mutate until reach the moment in which the union between their vertexes turns into the fitness gene. This procedure is shown in figure 6 where a cycle of mutagenesis for the N12 edge is done. In the case that the edges do not exist in the initial graph this operation will not be carried out so the plasmid won’t get the corresponding fitness field.

When the last step has already been done, if we selected only the plasmids with the fluorescence conferred by the five genes of specific representations corresponding to the edges of the graph, we could affirm that the HPP proposed has a positive solution. Remember we are facing the problem of figure 3A.

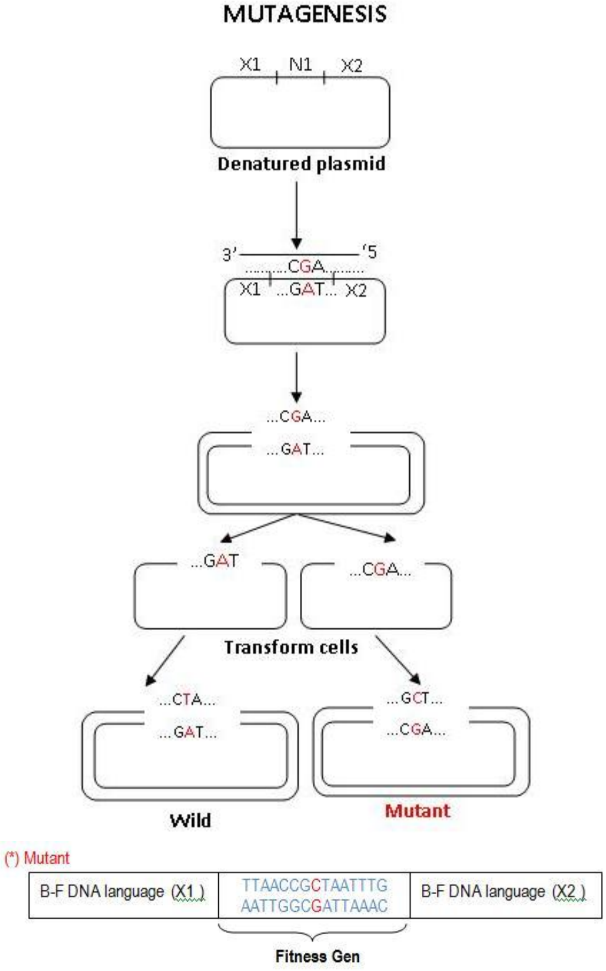

Figure 6.

In the same way, for that case in which a Hamiltonian Path does not exist in the given graph we would get a negative solution. That is due to the fact that we could notice after putting sequentially those bacteria which contain the vectors (once for each edge or protein) under fluorescent light of the absorption wavelength of the protein at issue that these ones do not emit.

In order to carry out these processes efficiently it is especially important to emphasize the use of bond-free languages for the codification of vertexes and edges providing stability and assuring the expected results during the formation and mutation steps of possible solutions of our problem.

4.2. GA-crossover operator by DNA shuffling

The ideas presented in this paper are proposed within the context of a genetic algorithm definition. This GA must be designed specifically for the resolution of the NP-complete problem known as TSP (Traveling Salesman Problem). The individuals of the population of this GA are codified using plasmids vectors in such a form that each one of these elements represents a possible solution of the cited problem.

Specifically each individual of the initial population must have been created following successive steps of a determined algorithm so each contains a specific sequence of a gene that includes information of several solutions. It could get all the possible combinations of edges of the graph in order to obtain all the possible solutions valid, invalid, complete and incomplete solutions of the problem.

The sequence of the each specific gene of the vectors belong to the initial population defines a random succession of cities, nodes in the graph which represents the problem.

During the implementation of the algorithm and after having evaluated the aptitude of each chromosome of the population, the best individuals are selected in order to be crossed and its descendants may be part of the next generation.

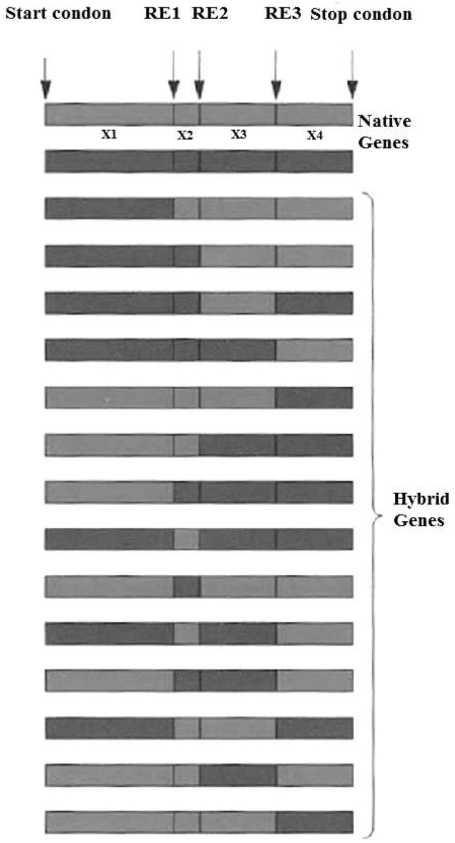

Figure 7. Generation of new genes by DNA shuffling

Selected chromosomes are crossed by the directed evolution technique DNA Shuffling as shown in Figure 7. The optimal solution, which is encoded in specific gene of a particular individual, is achieved following successive applications of the crossover operator in two individuals of each generation. One of the advantages of this technique is the possibility of introducing specific mutations in the genes of individuals selected for allowing achieve crossover areas of the space of search which are not covered by the individuals of the initial population.

5. CONCLUSIONS

This work presents theoretical approaches for solving complex combinatorial problems based on actual laboratory techniques. Using the well-known plasmids properties like information storage and massive replication it could encode NP-complete problems in order to obtain a solution by reducing the execution time. This proposal opens up a new way in the design of future experiments based on in-vitro directed evolution techniques in order to solve hard computational problems.

6. REFERENCES

[1]J.H.Holland, "*Adaptation in Natural and Artificial Systems*", MIT Press, 1975.

[2] M Mitchell, S Forrest, JH Holland, "*The Royal Road for Genetic Algorithms: Fitness Landscapes and GA Performance*", MIT Press, 1992.

[3]John Koza, "*Genetic Programming: A Paradigm for Genetically Breeding Populations of Computer Programs to Solve Problems*", STAN-TR-CS 1314, 1990.

[4]WP Deng, JA Nickoloff, "*Site-directed mutagenesis of virtually any plasmid by eliminating a unique site*", Analytical Biochemistry Volume 200, Issue 1, 1992.

[5]Leonard M. Adleman, "*Molecular Computation of Solutions to Combinatorial Problems*". Science (journal) 266 (11): 1021-1024, 1994.

[6]WPC Stemmer, "*DNA shuffling by random fragmentation and reassembly: In vitro recombination for molecular evolution*", PNAS October 25, 1994 vol. 91 no. 22 10747-10751, 1994.

[7] Richard J.Lipton, "*Using DNA to solve NP-Complete Problems*", Science, 268:542-545, 1995.

[8] Samir W. Mahfoud, David E. Goldberg,"*Parallel Recombinative Simulated Annealing: A Genetic Algorithm*", Parallel Computing 21(1995) 1-28, 1995.

[9] T. Head, G. Rozenberg, R.S. Bladergroen, C.K.D. Breek, P.H.M. Lommerse, H.P. Spaink, "*Computing with DNA by operating on plasmids*" BioSystems 57 (2000) 87-93, 2000.

[10] Kenichi Wakabayashi, Masayuki Yamamura .Natural Computing, "*A Design for Cellular Evolutionary Computation by using Bacteria*", Vol. 4, No. 3. (September 2005), pp. 275-292, 2005.

[11] Wenbin Liu, Xiangou Zhu, Guandong Xu, Quiang Zhang and Lin Gao "*A DNA based evolutionary algorithm for the minimal set cover problem*", Volume 3645/2005, Advances in Intelligent Computing, 2005.