Hybrid Crossover – Mutation Pair for Genetic Algorithm in Solving Fuzzy Shortest Path Problem – Predominant and Subordinate Ants

V. Anusuya[1], R. Kavitha[2]

[1] P.G & Research Department of Mathematics, Seethalakshmi Ramaswami College, Tiruchirappalli-620002, TamilNadu, India

[2] Department of Mathematics, Chevalier T.Thomas Elizabeth College for women, Chennai-600011, TamilNadu, India

Abstract: ***The reasons behind the evolution of fuzzy shortest path problem are, finding the path of least cost from source vertex to the destination vertex in the graph G=*$\{V,E\}$*. Fuzzy shortest path problem comprises of fuzzy numbers as parameters and here generalized trapezoidal fuzzy numbers and their characteristics are used. In order to upgrade the optimization, evolutionary optimization is often used and hence Genetic Algorithm (GA) is packed with Ant Colony Optimization (ACO) for the better optimization. Our objective of the research is to hybrid each and every individual genetic operator with ant. In this paper, we took mutation and crossover operators to hybrid, not only for proposed problem and also wherever in the Genetic Algorithm (GA) and network topology combination. The proposed methodology hybrids the characteristics of ants so called predominant and subordinate ants with the conventional operator in which, is a first experiment ever in the history of hybridization with the best of our knowledge. The most used crossover and mutation operators are reviewed and the proposed is compared. The implementation of proposed and conventional methods is carried out in MATLAB and experimental result explains the importance of crossover and mutation operators in genetic algorithm and also the effectiveness of the proposed hybridization in the convergence and time complexity of the algorithm.***

Keywords: Genetic algorithm, ant colony, generalized trapezoidal fuzzy number, hybridization, crossover, mutation, shortest path problem

2010 Mathematics Subject Classification: 03B52, 03E72

1. Introduction

The shortest path problem in which, isfinding the shortest path is very common and widely studied on graph theory and optimization areas. To achieve the best path, there are many algorithms which are more or less effective; depending on the particular case. Shortest path problems play an important role in routing messages efficiently in networks. Each method has got independent merit of its own address and different types of path searching in different situations. These algorithms of path searching are not always based on precise data. The reasons behind the evolution of fuzzy shortest path problem from the shortest path problem, finding the path of least cost from source to the destination in the graph G=$\{V,E\}$, where V representing the number of vertices and E representing number of edges in the graph, are the uncertainties of real values for the parameters of the paths in real time applications. It has various applications such as robotics, scheduling, communication, transportation, VLSI design, routing and mapping in which the shortest path problems are applied importantly. So to deal with uncertainty in searching, we feel that non-classical logic that is fuzzy logic will be the appropriate tool. Blue et al. [4] gives taxonomy of network fuzziness that distinguishes five basic types combining fuzzy or crisp vertex sets with fuzzy or crisp edge sets and fuzzy weights and fuzzy connectivity.

The fuzzy shortest path problem was first analyzed by Dubois and Prade. They utilized the conventional shortest path algorithms, to treat the fuzzy shortest path problem. Klein proposed a dynamic programming recursion-based fuzzy algorithm. Lin and Chen found the fuzzy shortest path length in a network by means of a fuzzy linear programming approach. Fuzzy shortest path problem comprises of fuzzy numbers as parameters and here generalized trapezoidal fuzzy numbers and their characteristics are used. The edges are represented by generalized trapezoidal fuzzy numbers and fuzzy ranking and fuzzy distance measure can be used as parameters of finding shortest path.

In order to upgrade the optimization, evolutionary optimization is often used and hence Genetic Algorithm (GA) is packed with Ant Colony Optimization (ACO) for the better optimization. Genetic Algorithm (GA) is the most powerful among the optimization methods which involves 'natural selection' and the survival of the best individual to the next generation. The ACO technique is impressed by real-ant-colony observations. It is a multiagent approach that was originally projected to resolve troublesome discrete combinatorial- optimization issues, like the traveling salesman problem (TSP) [6], [7]. In some studies, completely different ACO models were applied to fuzzy system design problems [11], [12].ZainudinZukhri [16] proposed hybrid ant based genetic algorithm and compares the results obtained from genetic and proposed algorithm. Cauvery [5] proposed mobile agents in genetic algorithm where ants are used as mobile agents. Shang Gao[13] proposes a novel ant colony genetic algorithm in which genetic and ant colony algorithm is mixed up and different mutation operation is carried out to select the best outcome.
Alessandro et. al. [2] review various crossover and selection process and discusses its influence in Multiprotocol Label System (MLS) allocation problem. They also compare the selection – crossover pair and find the best pair and worst

pair in terms of cost effective and execution time that deals with number of generations. Our objective of the research is to hybrid each and every individual genetic operator with ant. In this paper, we took mutation and crossover operators to hybrid, not only for proposed problem and also wherever in the Genetic Algorithm (GA) and network topology combination. The proposed methodology hybrids the characteristics of ants so called predominant and subordinate ants with the conventional operator in which, is a first experiment ever in the history of hybridization with the best of our knowledge. The most used crossover and mutation operators are reviewed and the proposed is compared in the following sections.

This paper is organized as follows. In section 2, explains the properties of generalized trapezoidal fuzzy numbers and describes the distance measure of generalized trapezoidal fuzzy numbers. Section 3 briefs the network terminology. Section 4 explains the approach of Genetic Algorithm (GA). Section 5describes the proposed predominant and subordinate ant hybridization used in crossover and mutation operation of the Genetic Algorithm (GA) along with the algorithm. In section 6, numerical example along with the example calculation is given. Section 7 deals with the results and discussion. And paper ends with the conclusion and future enhancement in section 8.

2. Basic Definitions

The basic definitions of some of the required concepts are reviewed in this section.

2.1 Fuzzy set

Let X be an universal set of real numbers R, then a fuzzy set is defined as A= {[x,$\mu_{\tilde{A}}$(x)], x ϵ X}.This is characterized by a membership function: X ⟶ [0 1], Where, $\mu_A(x)$ denotes the degree of membership of the element x to the set A.

2.2 Characteristics of generalized fuzzy number

A fuzzy set $\tilde{A}$ which is defined on the universal of discourse R, is known to be generalized fuzzy number if its membership function has the following characteristics

a) $\mu_A : R \to [0,1]\ is continuous.$
b) $\mu_A(x) = 0\ for all x \in (-\infty, a] \cup [d, \infty).$
c) $\mu_A(x) is strictly increasing on [a,b] and strictly decreasing$
d) $\mu_A(x) = w, for all x \in [b,c], where\ 0 < w \le 1.$

2.3 Membership function of generalized trapezoidal fuzzy number

A generalized trapezoidal fuzzy number $\tilde{A}$ = (a,b,c,d;w) is known to be a generalized trapezoidal fuzzy number, if its membership function is given by

$$\mu_{\tilde{A}}(x) = \begin{cases} \frac{w(x-a)}{(b-a)} & a \le x \le b \\ 1 & b \le x \le c \\ \frac{w(x-d)}{(c-d)} & c \le x \le d \end{cases}$$

Let $\tilde{A} = (a,b,c,d;w)$ be a generalized trapezoidal fuzzy number then

a) $R(\tilde{A}) = \frac{w(a+b+c+d)}{4}$, b) M$(\tilde{A}) = \frac{w(b+c)}{2}$, c) divergence $D(\tilde{A}) = w(d-a)$, d) Left spread $LS(\tilde{A}) = w(b-a)$, e) Right spread $RS(\tilde{A}) = w(d-c)$

2.4 Fitness function

Jahantigh [14] described the relation between generalized trapezoidal fuzzy number (a,b,c,d;w) and trapezoidal fuzzy number (a,b,c,d) that the trapezoidal fuzzy number has value of w=1 whereas generalized trapezoidal fuzzy number has the range 0≤w≤ 1 and also described the relation between generalized triangular (a,p,q;w) and trapezoidal (a,b,p,q;w) fuzzy numbers in which trapezoidal is equivalent to trapezoidal having a=b.

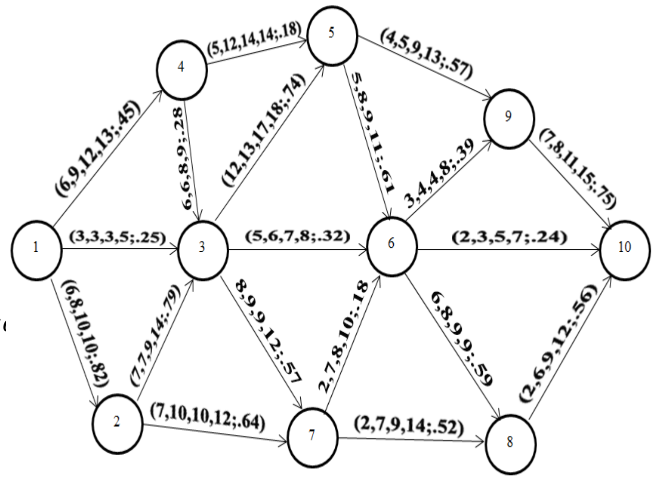
The distance measure between the generalized trapezoidal fuzzy numbers $\tilde{A}$ (a_1,b_1,c1,d1;w_1) and $\tilde{B}$ (a_2,b_2,c_2,d_2;w_2) using centroid points (α, β) of $\tilde{A}$ is given by [6]

$$f_d(\tilde{A},\tilde{B}) = \max\{|\alpha_{\tilde{A}} - \alpha_{\tilde{B}}|, |\beta_{\tilde{A}} - \beta_{\tilde{B}}|, |R(\tilde{A}) - R(\tilde{B})|, |LS(\tilde{A}) - LS(\tilde{B})|, |RS(\tilde{A}) - RS(\tilde{B})|\}$$

where $\alpha = \frac{1}{3}\left[a_1 + a_2 + a_3 + a_4 - \frac{a_4 a_3 - a_1 a_2}{(a_4+a_3)-(a_1+a_2)}\right]$ and $\beta = \frac{1}{3}\left[\frac{a_3 - a_2}{(a_4+a_3)-(a_1+a_2)}\right]$

3. Network Terminology

Consider the directed network G (V, E) consisting of a finite set of vertices V={1,2……n} and a set of m directed edges E $\subseteq V \times V$. Each edge is denoted by an ordered pair (i,j) where i, j∈ v and i≠ j. In this network, we specify twovertices namely source vertex and the destination vertex. $\widetilde{d_{ij}}$ denotes the generalized trapezoidal fuzzy number associated with the edge (i,j). The fuzzy distance along the path P is given in section 2.4.



4. Genetic Algorithm

Genetic Algorithm (GA) is a type of Evolutionary Algorithm (EA) which is based on the natural selection phenomenon. GA usually has an analogy to the randomness in solving a problem. It is comprised of generations where children are produced by the mating of the parents with genetic operators. Selection and reproduction to produce efficient generation is based on the random procedures, known to be natural selection.

4.1 Representation of an individual (chromosome)

Each chromosome is represented in integer representation and it is also important which represents the solution in the generations. The representation defines the path traversed and indirectly refers the fuzzy fitness of the chromosome. The number of integer used in representing chromosome various between the vertex visited and maximum to the number of vertices in the network graph G={V, E}. The vertex visited is represented by its own vertex number the vertex which is not visited is represented.

4.2 Population initialization

The initial population is generated randomly in usual GA and each chromosome represents the collection of edges which are represented by generalized trapezoidal fuzzy numbers explained in previous sections. The default population size 20 is used.

4.3 Selection operation

Selection operation is used in initialization process and parent selection for crossover operation. Various selection operations involve Roulette wheel selection, Random selection, Rank selection, Tournament selection and Boltzmann selection [14].
Here we choose distance measure considering rank, divergence, left spread and right spread of generalized trapezoidal fuzzy numbers explained in previous section as the fitness function. As the fuzzy shortest path problem is considered, minimum will survive by the selection the chromosome values.

The distance measure between every two edges is calculated and obtained distance is summed up throughout the path. The chromosome having minimum distance (fitness) will have more priority to be selected in the generation.

4.4 Crossover operation

Crossover operator mates two parent chromosomes and produces children which comprise the essence of two parent chromosome mated. Crossover operation is mainly categorised into two single point and multi point crossover . The single point crossover has single crossover site whereas multi point crossover has more than single crossover site. There are also some advanced multipoint crossover methods [14].

4.5 Mutation operation

The conventional mutation operator performs the minute changes of the reproduced child randomly under a certain rate which undo the degradation of the population due to crossover operation.

There were many mutation operations for real integers. Here we choose integer mutation that may be uniform, insertion, interchanging, boundary,non – uniform and others [3].

4.6 Elitism

Elitism in the genetic algorithm helps to obtain quality of the generations. It guaranties the health factor of the chromosomes that should not degrade from one generation to the next generations.

According to elitism, the reproduced individual is checked that its fitness should be less than the individual having maximum fitness in the generation. The reproduces individual which met these criteria will able to proceed to the next generation.

4.7 Termination condition

Termination condition produces the optimal solution through the convergence. Mostly termination condition will be the maximum number of generations. Other conditions are the idealness of the chromosomes in the generation. In order to test the algorithm, maximum number of generations can be used as termination condition which clearly represents the convergence of the algorithm.

Here, idealness of the chromosomes is considered as termination condition because of the usage trapezoidal fuzzy numbers and uncertainty in real numbers. When no change in the optimal fitness (minimal) and the idealness of the chromosomes in generations for at least 5 generations, then the algorithm reaches the termination condition.

5. Crossover and mutation

The objective of crossover and mutation operation is to perform the reproduction of the next generation crossover and it is the major operation that leads to the evolution of the optimization of the results and it has to keep tracked in order to upgrade the algorithm. The various most used crossover and mutation methods are reviewed below.

5.1 **One-point crossover (OPC)**

One-point crossover consists of single crossover point. A crossover point is a split point where both parental chromosomes combine to produce a child chromosome. These points are always a random segment that ranges between the lengths of parental chromosome. Obviously child chromosome is reproduced by appending chromosomes after the crossover point of one parent after the chromosomes till the crossover point of another parent. The successor and predecessor of the parental chromosomes are selected randomly and also the crossover point. Fig 5.1 briefly explains the one-point crossover.

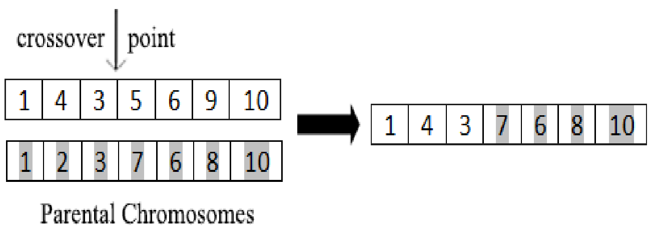


Figure 5.1: One-point crossover operation

5.2 Two-point crossover (TPC)

Two-point crossover is as same as one-point which consists of two crossover point where one belongs to the start and another to the end. A crossover point is a split point where both parental chromosomes keep in touch with each other to produce a child chromosome. The child chromosome is reproduced by appending chromosomes after the one crossover point of one parent after the chromosomes till the second crossover point of another parent and follows by the chromosomes after the second crossover point of same parent. The successor and predecessor of the parental chromosomes are selected randomly and also the crossover points. Fig 5.2 briefly explains the two-point crossover.

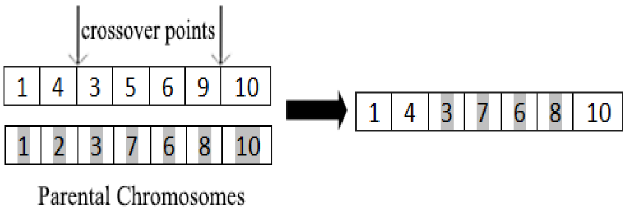


Figure 5.2: Two-point crossover operation

5.3Uniform crossover (UC)

The uniform crossover has different functionality between bit and integer chromosomes. In bit chromosomes, bit by bit masking is used to reproduce child from parental chromosomes. The uniformity is obtained by the criteria that when the bit is zero, the bit from the predecessor is chosen and when the bit is one, the bit from the successor is chosen. The successor and predecessor of the parental chromosomes are selected randomly. In case of integer, the average of the integer chromosome is calculated for successor parent. The uniformity is achieved by the criteria that when the integer of successor parent is greater than average, integer from predecessor parent is selected and when the integer of successor parent is lesser or equal to average, integer from predecessor parent is selected. Fig 5.3 briefly explains the uniform crossover.

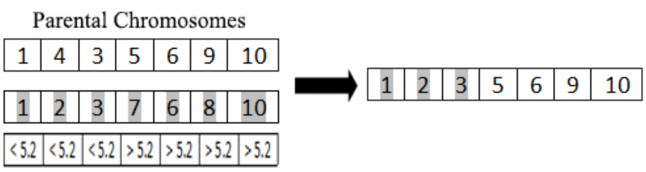


Figure 5.3: Uniform crossover operation

5.4**Uniform mutation (UM)**

Uniform mutation selects a random integer from the chromosomes that has to be mutated and replaced the selected integer with the random integer ranges between the user specific bounds. Here the user specific bounds lie in the range of vertex between the source and the destination vertex. Fig 5.4 briefly explains the uniform mutation.

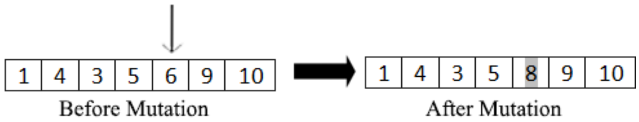


Figure 5.4: Uniform mutation operation

5.5Boundary mutation (BM)

Boundary mutation is same that of uniform in selection of random integer from the chromosomes that has to be mutated but differs only with the replaceable integer in which it would be user specific upper bound or lower bound. Fig 5.5 briefly explains the boundary mutation.

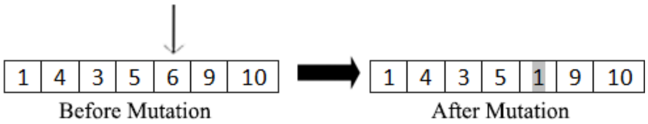


Figure 5.5: Boundary mutation operation

6. Proposed hybrid crossover and mutation operation

A crossover and mutation operation influences more in the reproduction of next generations of the genetic algorithm. The problem in the crossover and mutation is that the natural selection may leads to the unhealthy reproduction of next generation that leads to the increase of number of generations and affects convergence. Though natural selection is used, there should have certain criteria that should produce healthy reproduction that leads to the better convergence since reproduction is the source of evolution in genetic algorithm. The proposed method should overcome the identified problem.

The proposed hybridization comprises of Predominant Ant (PA) and Subordinate Ants (SAs) in which both has its unique characteristics in crossover and mutation operation.

6.1 Predominant Ant (PA)

Predominant Ant (PA) is the leading ant which regulates the crossover and mutation operations. The hybridization is obtained by guiding the genetic operation with the criteria by the ants. Predominant ant always keeps track on the outcomes of the crossover and mutation operation and moves forward for the result.

Predominant Ant (PA) moves on the continuous path without the help of the SubordinateAnts (SAs). Whenever the occurrence of discontinuity or crossover point, it tries to switch over to another parent. The tracking of SAs by the PA can be done on following ways.

a. When PA finds crossover point, it tries to connect the last visited vertex with the vertex after crossover point of another parent.
b. During the discontinuity of the path, PA seeks help from the SAs to validate the path between two vertices given by PA, where given vertices are previous vertex and the possible next vertex in another parent.

c. When SAs produce negative result for the above, the vertices for which the validation carried out are changed with new possible next vertex in same parent.
d. When SAs produce negative result for the above, PA randomly selects a vertex in the way that the path should be continuous.

Thus PA proceeds with the vertex visited and in case of discontinuity or crossover point, it switches over between the parent chromosomes till it reaches the destination node.

6.2 Subordinate Ant (SA)

Subordinate Ants (SAs) are underlying ants of PA which is capable of performing the tasks given by the Predominant Ant (PA). SAs donot have any sort of orderingas that of PA but possess very major role in the hybridization. Subordinate ants are not indeed to keep track on the outcomes as it does not know the source and reason of its work and simply follows the instruction of PA and revert back the need to the PA.

Predominant ant cannot make without the help of subordinate ants. SAs have the characteristics that its traversal is possible only on the continuous path. The size of the SAs depend on the input vertices given by the PA where size is equal to the number of outgoing edges in the start vertex given by PA. The function of SAs can by explained as follows.

a. Get the start and stop vertex from the PA whenever it needs help.
b. The size of the SAs is determined by the number of outgoing edges in the start vertex given by PA.
c. SAs move towards the individual outgoing edge in which no two SAs move through the same outgoing edge from the start vertex given by PA.
d. The SAs has to reach the end vertex given by PA and hence edge towards end vertex is preferred.
e. The aim of the SAs is to reach the end vertex as soon as possible with least number of edges between the paths.
f. SAs should stop their traversal when they reach end vertex given by PA or the destination node of the graph.
g. SAs reaching the end vertex specified by PA are the results for PA. The requirement of the PA is based on the circumstance and hence SAs are always ready to provide any sort of information about particular SAs.

When the given task cannot be completed, that no path exists between the given vertices, PA will take care of the result by itself.

6.3 Hybrid crossover

Hybrid crossover comprises of Predominant Ant (PA) and Subordinate Ants (Ants) with their unique characteristics. The hybrid crossover works on the criteria that it should not produce unhealthy (discontinuous) chromosomes without affecting the behavioral principle 'natural selection' of the genetic algorithm.

PA traverses the path that has to be traversed alone to reach the desired destination vertex whereas SAs traverses to find the alternate paths that cannot be reached by the PA. It has to be assumed that both PA and SAs traverse the existing path alone.

Crossover can be obtained by the PA where it chooses one parental chromosomes and traverse along the path till it finds discontinuity or crossover point.When PA finds discontinuity or crossover point, it switches over to chromosome just after the crossover point of next parental chromosomes. As explained in PA and SAs, the alternative path can be found and results the continuous path after the crossover.

Algorithm

1. Get the two parental chromosomes from the selection operation of genetic algorithm.
2. PA chooses one parental chromosome and traverse along the path specified by that parental chromosome.
3. PA switches over from selected parental chromosomes to next parental chromosomes and vice versa when it finds discontinuity or crossover point.
4. When PA finds discontinuity during the switch over, it seeks help from the SAs to find the alternative path with the given start and end vertex given by PA.
5. The alternative path is found by SAs or PA with the procedure explained in section 6.1 and section 6.2.
6. This procedure continues till the destination vertex of the graph is reached.

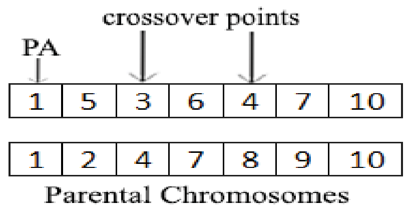


Figure 6.1: Crossover points in parental chromosomes

Fig 6.1 conveys the usage of two crossover points which are chosen randomly, ranges within the size of the parental chromosomes. The two parental chromosomes are selected which may or may not be continuous paths. PA should follow the criteria that whenever it reaches any discontinuous or crossover point, it has to switch over between the two parental chromosomes. Each and every movement of PA will be explained below diagrammatically for the better understanding.

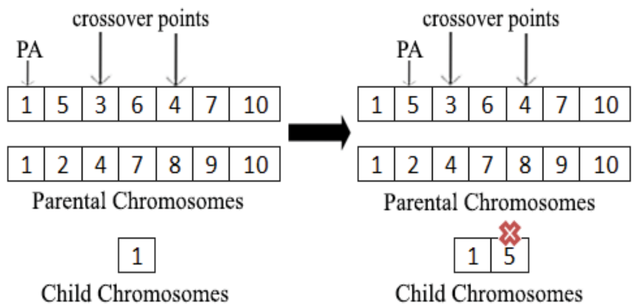


Figure 6.2: Traversal of PA between vertex 1 and vertex 5

Fig 6.2 explains that PA starts from the source vertex in the predecessor parent and moves towards next chromosome in the same parent. During the traversal, PA finds that there is no path between vertex 1 and vertex 5.

International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2013): 4.438

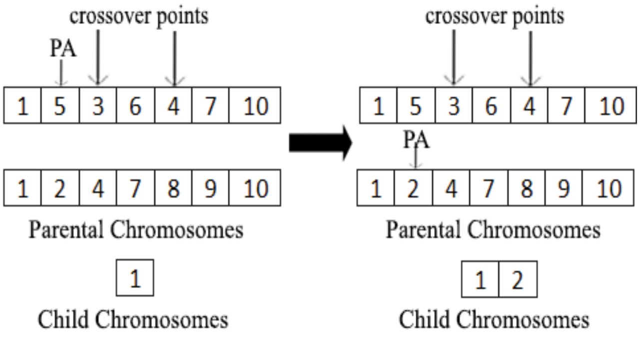


Figure 6.3: Traversal of PA between vertex 1 and vertex 2

Due to the discontinuity of the path between vertex 1 and vertex 5, PA switches over from predecessor parent to successor parent at the same place where it left from predecessor parent. The continuous path of vertex 2 is selected and traversed by PA.

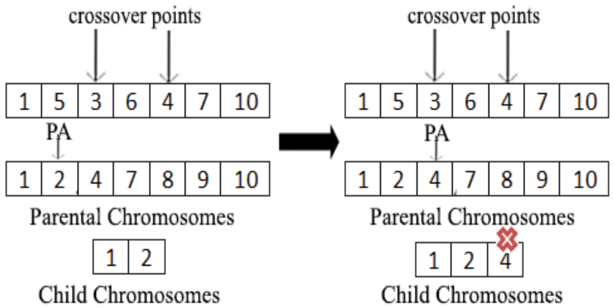


Figure 6.4: Traversal of PA between vertex 2 and vertex 4

PA which is in vertex 2, moves towards the next vertex 4 in the successor parent. During the traversal, PA finds that there is no path between vertex 2 and vertex 4.

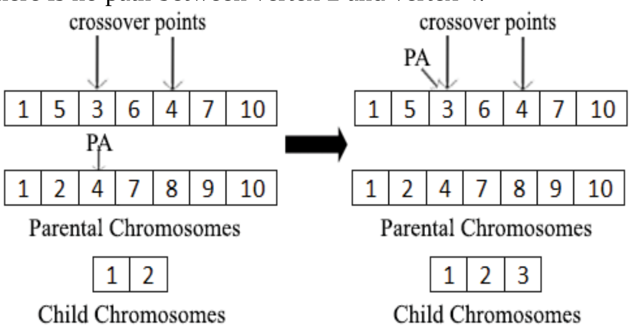


Figure 6.5: Traversal of PA between vertex 2 and vertex 3

Due to the discontinuity of the path between vertex 2 and vertex 4, PA switches over from successor parent to predecessor parent at the same place where it left from successor parent. The continuous path of vertex 3 is selected and traversed by PA.

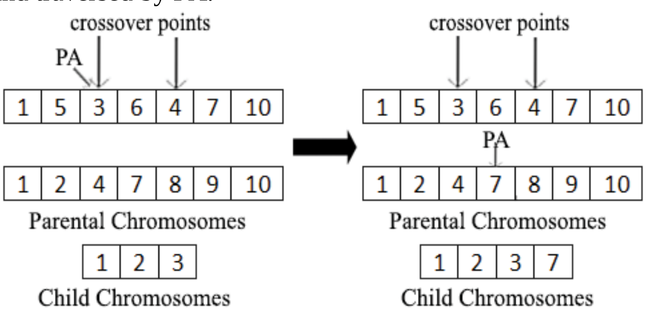


Figure 6.6: Traversal of PA between vertex 3 and vertex 7

PA which is in vertex 3 due to the crossover point, it switches over from predecessor parent to successor parent at the next place where it left from predecessor parent. The continuous path of vertex 7 in successor parent is selected and traversed by PA.

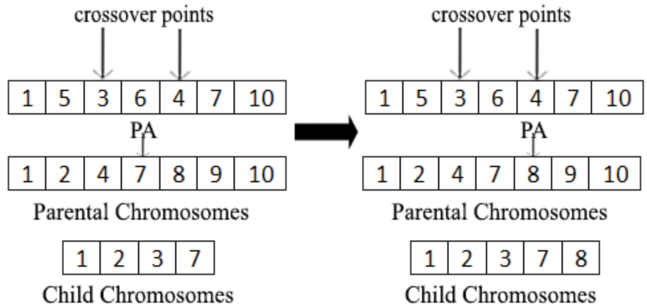


Figure 6.7: Traversal of PA between vertex 7 and vertex 8

Vertex 7 in the successor parent is the current position of PA and moves towards the next vertex 8 in the same parent. The continuous path of vertex 8 in successor parent is selected and traversed by PA.

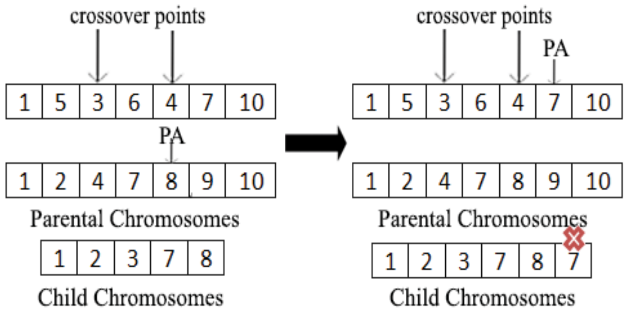


Figure 6.8: Traversal of PA between vertex 8 and vertex 7

PA which is in vertex 8 due to the crossover point, it switches over from successor parent to processor parent at the next place where it left from predecessor parent. During the traversal, PA finds that there is no path between vertex 8 and vertex 7.

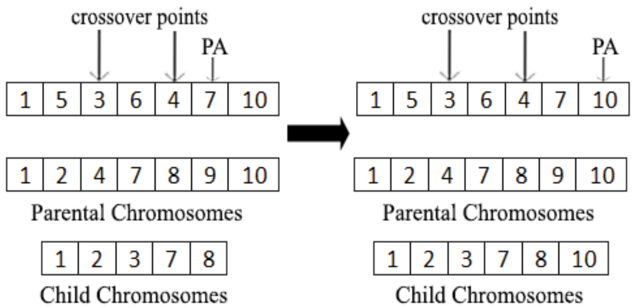


Figure 6.9: Traversal of PA between vertex 8 and vertex 10

Since crossover point, PA switches over between the parents and finds the discontinuity of the path between vertex 8 and vertex 7, PA seeks help from the SAs to find the alternate path wth the start vertex 8 to very next vertex 10. SAs return the direct path between vertex 8 and vertex 10 and thus vertex 10 is selected and traversed by PA.

In case there is no path between vertex 8 and vertex 10, PA has to switch over and checks for the availability of the path between vertex 8 and next vertex 9. The discontinuity of paths between vertex 8and vertex 9 leads to the PA to seek help from SAs from the start vertex 8 and end vertex 10 in successor parent. If there were no path between vertex 8 and vertex 10, PA then randomly selects a vertex from vertex 8 and traversed. Then it continues the procedure till the destination node is reached.

6.4 Hybrid mutation

Hybrid mutation works along PA and SAs with the criteria in producing healthy (continuous path) chromosomes for the next generation. The integer to be mutated is selected by the mutation value M_{vertex} which is given by

$$M_{vertex} = (Tot_{alter} - 1) * ran_{vertex}$$

Where M_{vertex} represents mutation value of a particular vertex, Tot_{alter} represents total number of alternative paths found by SAs from start vertex to end vertex given by PA and ran_{vertex} is existing vertex chosen randomly in the network graph.

PA traverses on the path specified by chromosomes and SAs helps to find the alternative path for the vertex given by PA to calculate M_{vertex} of given vertex. Mutation may or may not be takes place based on the condition that any vertex has $M_{vertex} > 0$.

Algorithm

1. Get the parental chromosomes to be mutated from crossover operator.
2. Initialize $M_{vertex} = 0$ for both source and destination vertex.
3. PA starts from the source vertex and traverses along the path of parent and initiates SAs to find Tot_{alter} for each vertex it traverses.
4. Calculate M_{vertex} for each vertex after the result Tot_{alter} of SAs and ran_{vertex} chosen by PA.
5. PA selects integer (vertex) from the chromosomes having maximum M_{vertex} and also $M_{vertex} > 0$ and no mutation takes place when the condition is not satisfied.
6. PA swaps any of the alternate paths given by SAs randomly with the chosen integer.

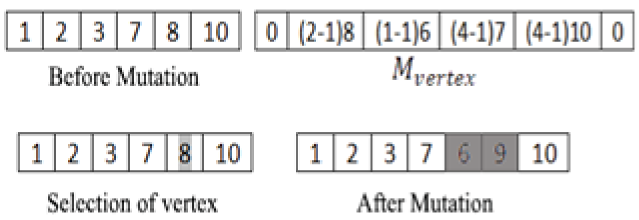


Figure 6.10: Hybrid mutation

Fig 6.10 explains the procedure of hybrid mutation. 'Before mutation' in the figure represents the individual obtained by the outcome of crossover. M_{vertex} represents the mutation value of each integer in the given individual where ran_{vertex} are chosen individually for each vertex by PA. The vertex 8 having greater mutation value (30) is selected as mutational integer represented in selection of vertex in the figure. Then PA seeks help from the SAs to find alternate paths between the start vertex 7 and end vertex 10. From the given path from SAs (7-6-10, 7-6-9-10 and 7-6-8-10), PA randomly selects the path and swaps with the mutational vertex 8 that is represented in 'After mutation' in the figure.

7. Numerical Example

We Consider the network G= {V, E} of co vertices (n=10). Every edge is represented by the generalized trapezoidal fuzzy number. The fitness, distance measure and properties of generalized trapezoidal fuzzy number in which described in previous chapters, are used to calculate. The representation and initialization of individual is already explained in the section 4.1 and 4.2. The fitness can be calculated using distance measure $f_d()$ (section 2.4) for proposed method. Let consider two edges A(3,3,3,5;.25) and B(5,6,7,8;.32) and the distance measure given in section 2.4 can be calculated using the centroid points, rank, Left spread and Right spread.

$$\alpha_A = 3.67 \ \alpha_B = 6.5 \ \alpha_A - \alpha_B = 2.83$$
$$\beta_A = 0.33 \ \beta_A = 0.42 \ \beta_A - \beta_A = 0.09$$
$$R(A) = .88 \ R(B) = 2.08 \ R(A) - R(B) = 1.2$$
$$LS(A) = 0 \ LS(B) = .32 \ LS(A) - LS(B) = .32$$
$$RS(A) = .50 \ RS(B) = .32 \ RS(A) - RS(B) = .18$$
$$f_d(\tilde{A}, \tilde{B}) = \max\{|\alpha_{\tilde{A}} - \alpha_{\tilde{B}}|, |\beta_{\tilde{A}} - \beta_{\tilde{B}}|, |R(\tilde{A}) - R(\tilde{B})|, |LS(\tilde{A}) - LS(\tilde{B})|, |RS(\tilde{A}) - RS(\tilde{B})|\}$$
$$= \max\{2.83, 0.09, 1.2, .32, .18\} = 2.83$$

Table 7.1: Sample calculation $f_d()$ of path 1-3-6-10 (continuous path)

Path	Next vertex	$f_d(\tilde{A}, \tilde{B})$(section 2.6)
1	3	0
1-3	6	0+2.833=2.833
1-3-6	10	2.833+2.214=5.047

Table 7.2: Sample calculation $f_d()$of path 1-4-5-8-10 (non continuous path)

Path	Next vertex	$f_d(\tilde{A}, \tilde{B})$(section 2.6)
1	4	0
1-4	5	0+2.475=2.475
1-4-5	8	2.475+(not exist)∞=∞
1-4-5-8	10	∞+3.6703 =∞

Thus continuous consequent execution of each operations of genetic algorithm leads to the reproduction of chromosomes which evolves with the principle of natural selection. The evaluation of chromosomes leads to the convergence of the result optimizes as 1-3-6-10 as shortest path between the source and the destination vertex.

8. Results and Discussion

The implementation is carried out in Matlab 8.1 (R_{2013a}) 32 bit student version. The other genetic operations such as chromosome representation, population initialization and selection are implemented as explained in the section 4. The hybrid crossover and mutation methods are used for the better convergence which concentrate on selecting best individual without affecting the randomness and the term 'natural selection' of the Genetic Algorithm (GA).

The network G={V,E} of 30 nodes with the edges of generalized trapezoidal fuzzy number is initialized. The algorithm is implemented as per the given description and demonstrated numerical calculation. Not only proposed method, also the methods that are mostly used as crossover and mutation implemented for checking the convergence and influence of proposed hybrid method in genetic algorithm.

The random function with time constraint is used to implement the random moves of both PA and SAs with unique behaviour. The random movement and the interactions between the PA and SAs are controlled by the adjacent matrix that describes the existence of the path between the vertices. The idealness of the best solution of about 10 consecutive generations is taken as termination criteria.

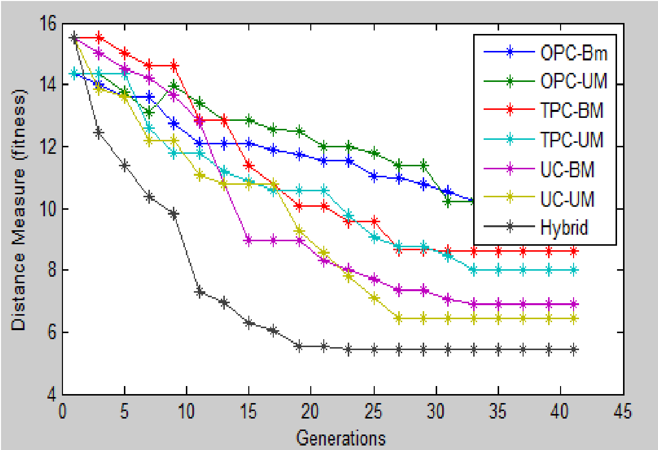


Figure 8.1: Comparison of various crossovers – mutations pair with the proposed hybrid method in terms of number of generations and optimum result

Fig 8.1 clearly represents the Comparison of various crossovers – mutations pair with the proposed hybrid method in terms of number of generations and optimum result. Some the pairs give optimized results and some pairs give worst results. From the comparison, the proposed hybrid crossover – mutation pair influences more in the convergence and provide extraordinary in the overall performance. Thus proposed hybrid crossover – mutation pair is concluded as best pair and OPC – BM is considered to be the worst pair.

9. Conclusion

The optimization of fuzzy shortest path problem is elaborately analyzed with the hybridization of Genetic Algorithm (GA) with the characteristics of ants. Various crossover – mutation pairs are reviewed and we propose a new hybrid crossover – mutation pair. The proposed methodology hybrids the characteristics of ants so called Predominant Ant (PA) and Subordinate Ants (SAs) with the conventional operator in which, is a first experiment ever in the history of hybridization with the best of our knowledge. The most used crossover – mutation pairs are reviewed and compared with proposed hybrid pair in terms of convergence. The implementation of proposed and conventional methods is carried out in MATLAB and experimental result explains the influence of crossover and mutation operators in genetic algorithm and also the effectiveness of the proposed hybridization in the convergence of the algorithm. The proposed hybrid pair is considered as best pair than the conventional crossover – mutation pairs.

References

[1] S. Abbasbandy, S. Salahshour., New results on the existing fuzzy distance measures, Iranian Journal of Fuzzy Systems, Vol. 10, No. 3, pp. 115-124, 2013.

[2] Alessandro Vivas Andrade, Luciano de Errico, Andr´e L. L. Aquino, Luciana Pereira de Assis,Carlos H. N. R. Barbosa, 'Analysis of Selection and Crossover Methods used by Genetic Algorithm-based Heuristic to solve the LSP Allocation Problem in MPLS Networks under Capacity Constraints,' EngOpt 2008 - International Conference on Engineering Optimization, Rio de Janeiro, Brazil, pp. 1-5, June 2008.

[3] Aluizio F. R. Araújoet. al., Multicast Routing Using Genetic Algorithm Seen as a Permutation Problem, Proceedings of the 20th International Conference on Advanced Information Networking and Applications, 2006.

[4] Blue, M., Bush, B. and Puckett, J., Unified approach to fuzzy graph problems, Fuzzy Sets and Systems 125, pp. 355–368, 2002.

[5] F.A Cauvery N K, S.B. Dr K V Viswanatha., Routing Algorithm using mobile agents and Genetic Algorithm, International Journal of Computer and Electrical Engineering, Vol. 1, No. 3, pp. 293-296, 2009.

[6] M. Dorigo, V. Maniezzo, and A. Colorni, "Ant system: Optimization by a colony of cooperating agents," IEEE Trans. Syst., Man, Cybern. B,Cybern., vol. 26, no. 1, pp. 29–41, Feb. 1996.

[7] M. Dorigo and L. M. Gambardella, "Ant colony system: A cooperative learning approach to the traveling salesman problem," IEEE Trans. Evol.Comput., vol. 1, no. 1, pp. 53–66, Apr. 1997.

[8] M. Dorigo and T. St¨utzle, Ant Colony Optimization. Cambridge, MA: MIT Press, Jul. 2004.

[9] M. J. Ebadi, M. Suleiman, Fudziah Bt. Ismail, A. Ahmadian, M. R. BaloochShahryari, and S. Salahshour., A new distance measure for trapezoidal fuzzy numbers, Hindawi Publishing Corporation, Mathematical Problems in Engineering, Volume 2013, Article ID 424186, pp. 1-4, 2013.

[10] M. A. Jahantigh, S. Hajighasemi., Ranking of generalized fuzzy numbers using distance measure and similarity measure, Int. J. Industrial Mathematics (ISSN 2008-5621), pp. 405-416, 2012.

[11] C. F. Juang, C. M. Lu, C. Lo, and C. Y. Wang, "Ant colony optimization algorithm for fuzzy controller design and its FPGA implementation," IEEE Trans. Ind. Electron., vol. 55, no. 3, pp. 1453–1462, Mar. 2008.

[12] M. Mucientes and J.Casillas, "Quick design of fuzzy controllers with good interpretability in mobile robotics," IEEE Trans. Fuzzy Syst., vol. 15, no. 4, pp. 636–651, Aug. 2007.

[13] Shang Gao, Zaiyue Zhang and Cungen Cao., A novel ant colony genetic hybrid algorithm, Journal of Software, VOL. 5, NO. 11, pp. 1179-1186, 2010.

[14] Sivanandam S.N. &Deepa S.N. (2008): Introduction to Genetic Algorithms, 2008.

[15] Zadeh, L.A., Fuzzy sets as a basis for a theory of possibility, Fuzzy Sets and System 1, pp. 3–28, 1978.

[16] ZainudinZukhri and Irving VitraPaputungan, "A hybrid optimization algorithm based on genetic algorithm and ant colony optimization," International Journal of Artificial Intelligence & Applications (IJAIA), vol. 4, no. 5, pp. 63-75, Sept. 2013.