Design a Smart and Intelligent Routing Network Using Optimization Techniques: Proceedings of ICDMAI 2018, Volume 2

Chapter *in* Advances in Intelligent Systems and Computing · January 2019

DOI: 10.1007/978-981-13-1274-8_23

Design a Smart and Intelligent Routing Network using Optimization Techniques

Joyce Yoseph Lemos[1],Abhijit R. Joshi[2] ,Manish M. D'souza[3], Archi D. D'souza[4]
[1]St. John College of Engineering and Management, Palghar, India
[2]D. J. Sanghvi College of Engineering, Mumbai, India
[3]Senior Web Developer, Media.Net Corporation, Mumbai, India
[4]Research Scholar, Illinois Institute of Technology, Chicago, U.S.
{ Joyce Yoseph Lemos, Abhijit R. Joshi, Manish M. D'souza, Archi D. D'souza}
lemosjoyce01@gmail.com, abhijit.joshi@djsce.ac.in ,dsouzamanish@gamil.com, Archidsouza18@gmail.com

Abstract. The Routing Problem (RP) is designed to find the minimal set of routes in order to deliver the services to a set of customers. There are many techniques available to solve RP problems such as Ant Colony Optimization (ACO), Honey Bee Optimization (HBO), Particle Swarm Intelligence, Genetic Algorithm (GA) and much more. Our aim is to consider the various RP techniques and compare the performance of each technique based on various parameters. To carry out this, we have considered the School Bus Routing Problem (SBRP), which ensures the on-time delivery of students to and from school by minimizing the total travel cost in terms of time and distance. The result of this research work helps to determine the best RP algorithm not only based on shortest path but also additional parameters such as pheromone level, optimum path level and distance.

Keywords: Routing Problem, Pheromone Trail, Vehicle Routing Problem, Ant Colony Optimization, Greedy Randomized Approach, Honey Bee Optimization

1 Introduction

The Routing Problem (RP) also referred as Vehicle Routing Problem (VRP) is a combinatorial optimization problem, which deals with transportation of goods from service point to customers [1]. The VRP problem is solved in two parts, firstly, by determining the set of stops to be visited and secondly, finding the set of minimal number of routes between source and destination by traversing the each route in routing network. The VRP problem considers the various parameters to do so, such as vehicle capacity, driver's riding time, time window of product delivery and minimum transportation cost.

The RP was first implemented at the end of fifties to solve the problem of transportation of gasoline from service station to customer point. At the same time some researchers were set the mathematical programming formulation and algorithmic approach to solve the same problem. The definition of RP states that initially the set of vehicles 'm' are located at a depot in order to deliver the fixed quantities of goods to a set of customers 'n'[2][8]. The RP problem is represented by determining the set of optimal routes when serving the set of customers by set of vehicles. The objective of RP is to minimize the overall transportation cost in terms of time and distance. The result of the RP problem is a set of routes starting from source to destination (which all begin and end in the depot), and which satisfies the constraint that all the customers are served only once. The transportation cost is depending on total travel distance and the required number of vehicles. The majority of the real world problems are often much more complex than the classical RP. Therefore in practice, the classical VRP problem considers the constraints such as vehicle capacity and time interval between the deliveries of services by ensuring that each customer has to be served only once. RP is hard combinatorial optimization problem which works only for small instances of problem. If the set of instances are increased, the RP does not guarantee the optimality of result.

From the last twenty years, the meta-heuristic approach has given the proper direction to solve the problems from VRP family. The School Bus Routing Problem (SBRP) was proposed in literature by Newton and Thomas (1969) [1] [2]. The SBRP was represented by Park and Kim (2010), as efficient scheduling of school buses in order to pick the students from various stops and delivers them to the school [3]. The constraints needs to satisfied for SBRP as maximum capacity of buses, maximum riding time of a student in a bus, and delivery time or time window to the school. According to the classification proposed by Desrosier et al. (1981), the SBRP consists of smaller sub-problems: data preparation, bus stop selection (student assignment to stops), bus route generation, school bell time adjustment, and route scheduling [4]. As described in Park and Kim (2010), in the data preparation step, the road network is formed, which consist of home, school, bus depot, and the origin-destination (OD) matrix. For a given road network, the bus stop selection step determines the location of stops, and the students are assigned to them. Thereafter, in bus route generation, the bus routes for a single school are generated. The school bell time adjustment and route scheduling steps determine the time window within the students has to be arrived to the school.

In most existing approaches in literature, steps in SBRP are considered separately and sequentially, although they are highly interrelated. In existing combinatorial optimization problems a single sub-problem or combination of them depends on set of instances and complexity of algorithm. For example, the bus route generation sub-problem is very similar to the VRP in which if each vehicle is considered separately, the problem becomes the well-known Traveling Salesman Problem (TSP), while the combined problem of bus stop selection and bus route generation falls into the class of Location-Routing Problems (LRP) [5]. The SBRP is a special case of the VRP in which, a set of 'n 'students has to be serviced by a set of buses. Also the problem considers the capacity of buses and since becomes the Capacitated Vehicle Routing Problem (CVRP), which is known to be NP-hard. The SBRP's main objective is to obtain the minimum set of routes to serve the students on time and to minimize the overall transportation cost as VRP. To achieve the good results, the problem is solved by applying various algorithms and determining which algorithm is best suited to solve the problem under this research work.

2 Related Work

In section details the literature survey of shortest path algorithms and optimization algorithms to carry out work in proposed methodology.

2.1 Dijkstra Algorithm

Dijkstra algorithm is used to find the shortest path between nodes in a graph (road network). It was proposed by computer scientist Edsger W. Dijkstra in 1956 [8]. Dijkstra algorithm considers only single node as "source" node and finds shortest paths from the source to all other nodes in the graph. Once all the paths from source to every other node are found, the shortest path is determined by the algorithm. Dijkstra algorithm also used to find the shortest path between single source and single destination. In road network, the nodes of graphs represent the cities and cost on each pairing edges of nodes represents the distances between the cities. For this, Dijkstra algorithm can be used to find the shortest path between one city and all other cities in the road network.

2.2 Bellman – Ford Algorithm

The Bellman–Ford algorithm is also a single source algorithm that computes shortest paths from a single source node to all of the other nodes in a weighted graph. It is slower than Dijkstra algorithm for the same problem, but more versatile, because it considers the negative weight of edges while computing the shortest path. The algorithm was proposed by Richard Bellman and Lester Ford in 1958 and 1956, respectively and hence the name "Bellman-Ford Algorithm" [9]. If a graph contains a "negative edge value", and that edge falls in the shortest path from source to destination, then the "negative cycle" is formed by one walk around the edge and the new path is generated by considering the cheapest positive value.

2.3 Genetic Algorithm

The Genetic Algorithms (GA) is a modern computer technique based on some ideas taken from biological evolution theory. In GA, initially a random or heuristic population set is generated. Then the cycles are repeated for number of instances in population set. GA also does the re-evaluation of cycles to determine the optimum solution. GA is sometimes termed as 'Evolutionary Algorithm' (EA) in which the main strategy is to find the optimum solution by using search operators such as natural selection, mutation and recombination to the population. GA is especially useful for problems having few, but not different solutions or problems having the exact solutions. In GA, the problems may not be having the exact procedure but the procedure estimation gives the optimum solution. This helps to eliminate some solutions and to accept another solutions i.e. one of the many good solutions [4].

2.4 Particle Swarm Optimization

Particle Swarm Optimization (PSO) works on the social behaviour of the groups of population in nature such as animal herds or bird flocking, or schooling of fish. PSO consists of a population called swarm and each member of the swarm is called a particle [10]. In PSO, the local and global optimum search is carried out on the behaviour of particles in swarm set. Particles changes and updates their position with respect to the distance between them and their neighbourhood. With this, the global search is performed to find the optimum solution. The optimal position of node on the graph is determined by updating the particle velocities. In every iteration, the fitness of each particle's position is determined by fitness measure and the velocity of each particle is updated by keeping track of two "best" positions.

2.5 Ant Colony Optimization

The Ant Colony Optimization (ACO) algorithm is a probabilistic technique based on the behaviour of ants, searching a path between their colony and a food source. In the natural world, ants' searches for their food and return to their colony by laying down pheromone trail on the path. If other ants find such a path, instead of travelling randomly they also follow the same path as pheromone is laid down on it. After some time, the pheromone starts evaporating which results in decreasing the likelihood of the path. If pheromone is not evaporated at all on the path, the paths chosen by first ant would be selected by all following ants [7] [10]. Thus, when one ant finds a good path from the colony to a food source, other ants are more likely to follow that path, resulting in selecting the single path by all ants.

2.6 Honey Bee optimization

The Honey Bee optimization (HBO) algorithm was proposed in 2005 based on the behavior of bees in natural world. This algorithm is divided into two categories, according to the bee's behavior in the nature, the foraging behavior and the mating behavior. Each candidate solution is considered as a food source (flower), and a set of population (colony) of 'n' agents (bees) is used to search the solution space. Each time an artificial bee visits a flower to evaluate its fitness function [9]. In HBO mating process, the set chromosomes are formed by probabilistically selecting the set of sub-solutions from the best solution. Once the set of chromosomes are formed the crossover operation is performed to generate the subset of child solutions. Mutation operation modifies one or more values in a chromosome from its initial state. After mutation the solution may change entirely from the previous solution.

2.7 Greedy Randomized Adaptive Search Approach

The objective of the Greedy Randomized Adaptive Search Procedure (GRASP) is to repeatedly select the samples of greedy solutions, and then use a local search procedure to refine them to local optima. During each iteration, the current solution is chosen as new current solution and best solution is determined based on the local optimum value between them. Greedy Randomized Approach is a meta-heuristic algorithm for combinatorial problems, in which each iteration consists of two phases: construction and local search [6]. The construction phase builds a feasible solution by investigating all solutions until local optimum is reached during the local phase.

3 System Methodology

The system mainly consist of two phases namely, pheromone evaporation & pheromone deposition. Firstly, the nodes are initialized with ants (number of buses required to pick up students from stops). The routes are generated by visiting the stops one by one. Once the particular route is visited the pheromone is deposited on that route & local pheromone trail is calculated. The process is repeated until all the routes are visited & finally the global pheromone trail is calculated, which gives the optimum route.

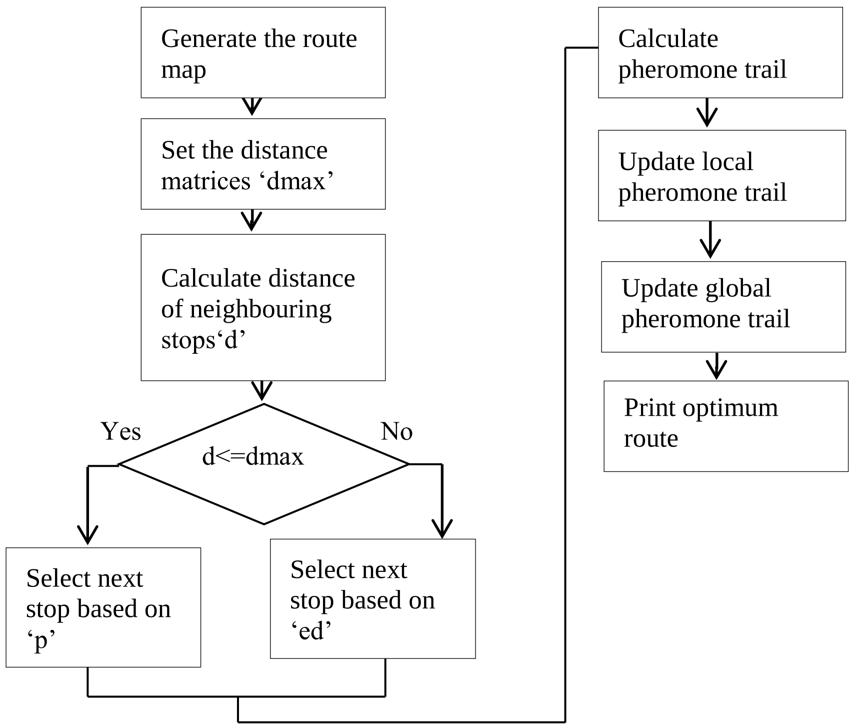

Figure 1: *Flowchart of Proposed Methodology*

The route map is generated, which consists of number of stops and number of students to be picked up from stops. The route is generated by visiting each stop. When the route is visited for first time the pheromone is deposited in that route and pheromone rate is increased in each subsequent visit for the same route. The distance between the nodes is calculated using shortest path algorithm and the values are stored in distance matrices. The maximum distance value from distance matrices is considered as 'dmax' and the weight of each edge between neighbouring nodes is calculated. If the distance'd' between neighbouring nodes is less than 'dmax' then next stop is selected using pheromone deposition rate and if it is not, then next stop is selected using Euclidean distance formula. Once the stop is selected, the pheromone trail is calculated for that path. If same path is selected in next iteration, update pheromone trail for the same. Once the local pheromone is updated, the global pheromone is calculated to find the optimum path.

4 Experimental Conditions

For the analysis carried out here the parameters of algorithms were considered: number of cycles (I), number of ants (K), Algorithm Parameter (q_0) and Beta Value (β). If one parameter is changed, the others remain constant.

4.1 Input Condition

Algorithm takes dataset Algorithm takes dataset from standard text (.txt) file. The user can provide any number of stops based on the requirement. The input to the algorithm will be number of stops followed by X-coordinate & Y-Coordinate to print the stop point on screen. The dataset can be randomly generated or taken from any standard benchmark dataset of TSP or VRP. It also considered that on each stop single student will get picked up. It will also not accept same stop location more than once. All stops have to be distinct.

4.2 Number of Cycles (I) & Number of Ants (K)

For the sensitivity analysis, we also used 50 and 200 buses. For the analysis, we considered I=50, I=90, I= 150, I=240 and I=300. Each bus should visit at least 50 stops (considering total numbers of buses are 10). All buses visits 50 stops with random points allocation for the values of I. Once the random allocation of stops on graph is completed, the local value for I is determined by adding the random point values of stops as follows

$$I_L = \sum_{i=1}^{n} R_{ij}$$

Where 'n' is the total number of stops and 'R_{ij}' is random point value for stop (random point of stop on screen). When I= 50 then I_L= 30.59511 (adding the random points for each stop on screen as R_{ij}) and I= 300 then I_L= 24.2742984 for Bus 1. Same process is applied for Bus 2, Bus 3 and Bus 4. As the number of buses increases the number of cycles required to visit the route decreases. From the local value of I, the global value for I is determined to find the total distance travelled by all buses for all the values of I. The global value for I is calculated as,

$$I_G = {\left(\sum_{j=1}^{K} I_L \times n\right)}\Big/_{2}$$

where 'K' is the total number of buses. For bus 1 the value for I_G= 3333.925. Similarly for Bus 2, Bus 3 and Bus 4 the global value for I is calculated. (See Table 1).

Table 1: *Cycle Sensitivity Analysis*

Bus No	**Distance Travel**	**Stops Visited**
Bus 1	3333.925	50
Bus 2	3117.45605	50
Bus 3	3094.432	50
Bus 4	3283.36175	50
Bus 5	3405.294554	50
Bus 6	3258.152934	50
Bus 7	3015.280347	50
Bus 8	3261.952611	50
Bus 9	3112.165453	50
Bus 10	3616.843016	50
Total	**32498.86**	**500**

4.2 Number of Ants/Buses (K)

For the analysis, we considered K=50, K=100, K= 150, and K=200. To calculate the value of total travelled distance with varying values of K and I, we have to consider the global value for I (I_G), which determines the total travelled distance with all the values for K. The local value for K is calculated as,

$$K_L = I_G \div K$$

Here 'K' value starts from 50,100,150 and 200. When K=50, the value of K_L for Bus 1, Bus 2, Bus 3 and Bus 4 is 66.6785, 62.3491, 61.88864 and 65.6672 respectively. As the number of buses increases, the distance travelled by each bus decreases with respect to the value of I_G. That is with K= 200 the value of K_L for Bus 1 Bus 2, Bus 3 and Bus 4 is 16.6696, 15.5872, 15.4721 and 16.4168 respectively . Then the global value for K is calculated as

$$K_G = \sum_{i=1}^{K} K_L \times K$$

Once the global value for K is calculated, finally, the distance travelled by all buses with varying values of I is determined. The K_G value for Bus 1, Bus 2, Bus 3 and Bus 4 is 138.9135, 129.8939, 128.9346 and 136.8067 respectively. After observing Tables 1 and 2, we can say that with more number of buses the same path can be repeatedly selected so distance covered by buses is less as compared to distance coverage by more number of cycles. (See table 2)

Table 2: *Ant Sensitivity Analysis*

Bus No	Distance Travel	Stops Visited
Bus 1	138.9135	50
Bus 2	129.8939	50
Bus 3	128.9346	50
Bus 4	136.8067	50
Bus 5	140.4039	50
Bus 6	122.3215	50
Bus 7	131.9886	50
Bus 8	125.9036	50
Bus 9	130.6015	50
Bus 10	133.9577	50
Total	**1319.726**	**500**

4.3 Algorithm Parameter (q_0)

The parameter also tested with values of 0.8 and 0.95. The algorithm parameter is used to find the pheromone level deposited on the visited route. It is calculated as,

$$g(p) = \frac{(k_G \times q_0)}{K}$$

Table 3: *Algorithm Parameter Sensitivity Analysis*

q_0= 0.8			q_0 = 0.95		
Bus No	**Distance Travel**	**Stops**	**Bus No**	**Distance Travel**	**Stops Visited**
Bus 1	11.11038	50	Bus 1	13.1967	50
Bus 2	10.391512	50	Bus 2	12.3399	50
Bus 3	10.314768	50	Bus 3	13.57206	50
Bus 4	10.9445	50	Bus 4	14.4007	50
Bus 5	11.6748	50	Bus 5	13.7895	50
Bus 6	11.4352	50	Bus 6	14.6757	50
Bus 7	10.5320	50	Bus 7	14.76456	50
Bus 8	11.45432	50	Bus 8	13.8789	50
Bus 9	11.60768	50	Bus 9	13.7687	50
Bus 10	10.5535	50	Bus 10	14.6678	50
Total	**110.0187**	**500**	**Total**	**139.0545**	**500**

Where K_G is global value for K, q_0 is either 0.8 or 0.95 and K is the total number of buses. When q_0= 0.8 the values are 11.11038, 10.391512, 10.314768, 10.9445 and with q_0=0.95 the values are 13.1967,12.3399,13.57206,14.4007 for Bus 1, Bus 2, Bus 3 and Bus 4 respectively. With q_0= 0.8 the distance travelled by each bus is less (i.e. 110.0187) than the higher value of q_0 i.e. 0.95 (i.e. 139.0545) (See Table 3). So with less value of q_0 the chances of selecting same path are less and distance coverage is also less as compared to high value of q_0.

4.4 Beta (β)

Looking at parameter β, table 5.3.4 presents the results of two sensitivity experiments. We have tested the value for β= 3 and β=8.

$$\rho = \frac{g(p)}{\beta}$$

Where 'g (p)' is algorithm parameter value for each bus and β is either 3 or 8. When the values are 3.7034, 3.4638, 3.4382, 3.6461 and with β=8 the values are 1.6495, 1.5424, 1.6965, 1.8 for Bus 1, Bus 2, Bus 3 and Bus 4 respectively. With β= 3 the distance travelled by each bus is more (i.e. 35.63919) than the higher value of β i.e. 8 (i.e. 16.32859) (See Table 4).

Table 4: *Beta Sensitivity Analysis*

β= 3			β=8		
Bus No	**Distance Travel**	**Stops Visited**	**Bus No**	**Distance Travel**	**Stops Visited**
Bus 1	3.7034	50	Bus 1	1.6495	50
Bus 2	3.4638	50	Bus 2	1.5424	50
Bus 3	3.4382	50	Bus 3	1.6965	50
Bus 4	3.6481	50	Bus 4	1.8	50
Bus 5	3.7878	50	Bus 5	1.878	50
Bus 6	3.00898	50	Bus 6	1.35454	50
Bus 7	3.5251	50	Bus 7	1.4068	50
Bus 8	3.87687	50	Bus 8	1.2342	50
Bus 9	3.8654	50	Bus 9	1.8899	50
Bus 10	3.32154	50	Bus 10	1.87675	50
Total	**35.63919**	**500**	**Total**	**16.32859**	**500**

4.5 Pheromone Evaporation Co-efficient (p)

The next analysis was carried out for parameter p. For this analysis, we choose the values 0.3 and 0.7. The value of pheromone evaporation co-efficient determines the evaporation speed of the pheromone trace. It is calculated as

$$P = g(p) \times \rho \times p$$

Where '*g (p)*' is algorithm parameter value for each bus ρ is beta value for each bus and 'p' is either 0.3 or 0.7. When p= 0.3 the values 12.3438, 10.7932, 10.63926, 11.97798 and with p=0.7 the values are 15.2375, 13.3231, 16.1174, 18.1448 for Bus 1, Bus 2,Bus 3 and Bus 4 respectively. With p= 0.3 the distance travelled by each bus is less (i.e. 117.7865) than the higher value of p i.e. 0.7 (i.e. 158.1579) (See Table 5).

Table 5: *Pheromone Evaporation Sensitivity Analysis*

p= 0.3			p=0.7		
Bus No	**Distance Travel**	**Stops Visited**	**Bus No**	**Distance Travel**	**Stops Visited**
Bus 1	12.3438	50	Bus 1	15.2375	50
Bus 2	10.7982	50	Bus 2	13.3231	50
Bus 3	10.63926	50	Bus 3	16.1174	50
Bus 4	11.97798	50	Bus 4	18.1448	50
Bus 5	13.26654	50	Bus 5	18.1276	50
Bus 6	10.3224	50	Bus 6	13.9151	50
Bus 7	11.13789	50	Bus 7	14.7678	50

Bus 8	13.32207	50	Bus 8	15.8799	50
Bus 9	13.4607	50	Bus 9	18.76687	50
Bus 10	10.51761	50	Bus 10	13.8778	50
Total	**117.7865**	**500**	**Total**	**158.1579**	**500**

5 Analysis of ACO, HBO and Greedy Randomized Approach

For the analysis carried out here, the parameters considered are: Pheromone level, Distance and Optimum path level. Pheromone level is defined as how many times the same path is selected by multiple numbers of buses. Optimum path level determines the ratio of same path selection to the number of buses. Distance parameter calculates the total distance travelled by each bus.

In ACO technique, when the numbers of buses are increased, the pheromone level obtained by buses also gets increased. In ACO, if the numbers of buses are more, then the same path can be selected by more number of buses. Also in ACO, the optimum path selection is proportional to the ratio of pheromone level to the number of buses. So when the pheromone level increases the optimum path level also decrease. ACO mainly considers the highest value of q_0. In ACO, the pheromone level is increased with decreasing value of optimum path level. For the highest value of q_0 i.e. 0.95, the distance travelled by each bus is also more compared to the minimum value of q0 i.e. 0.8. On the other hand, if value of q_0 increases that is with more number of buses, the value of β (optimum path level) decreases. (See Figure 2)

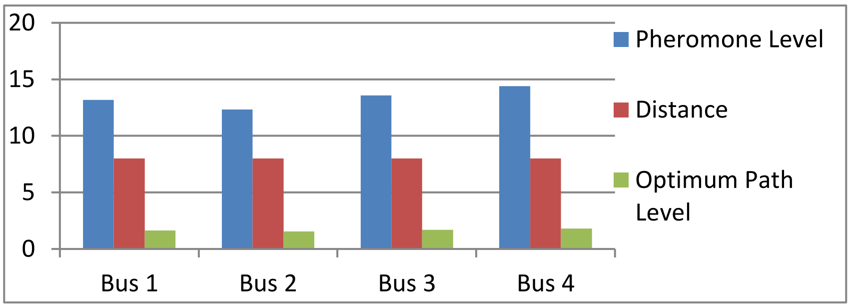

Figure 2: *ACO output with parameter values*

When the bus 1 starts at one particular point and its next stop is assigned randomly (as no other stops visited yet). In the next subsequent phase, the next stop can be chosen based on pheromone evaporation rate. If it is more, it means the path is selected more than once, and then the next stop is selected, which is not visited. The pheromone level is increased for that it is calculated with q0=0.95 and the optimum path level decreases, it is calculated with β=8. The distance parameter remains constant with changing values of pheromone level and optimum path, as it is the ratio of pheromone level to optimum path level. For Bus 1 the pheromone level is 13.167 with less optimum path value which is 1.6495. (Same for Bus 2, Bus 3 and Bus 4). But the distance value remains constant as 8. (See Table 6)

Table 6: *ACO parameter values*

Number of buses	Pheromone Level	Distance	Optimum Path Level
Bus 1	13.1967	8.00042	1.6495
Bus 2	12.3399	8.00045	1.5424
Bus 3	13.57206	8	1.6965
Bus 4	14.4007	8.003	1.8

In HBO technique, when the numbers of buses are increased the pheromone level decreased but optimum path level increases. As in HBO when numbers of buses are decreased, possibility of selecting the same path repeatedly gets increased. HBO mainly considers the lowest value of q_0. In HBO the pheromone level is decreased with increasing value of optimum path level. For the lowest value of q_0 i.e. 0.8, the distance travelled by each bus is also less compared to the maximum value of q0 i.e. 0.95. On the other hand, if value of q_0 decreases, that is with less number of buses, the value of β (optimum path level) increases. It means that in HBO, the low pheromone value results in high optimum path selection. (See Figure 3)

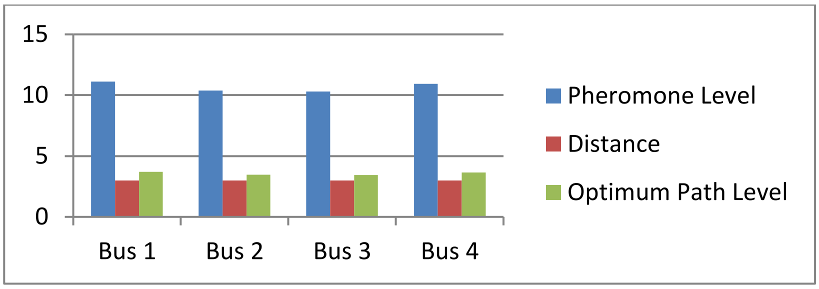

Figure 3: *HBO output with parameter values*

A bus 1 start at one particular point and its next stop is assigned randomly (as no other stops visited yet). In the next subsequent phase, the next stop can be chosen based on pheromone evaporation rate. If it is more, it means the path is selected more than once and has large visiting ratio, so the path is selected in next subsequent phase. The pheromone level is decreased for that it is calculated with q0=0.8 and the optimum path level increases, it is calculated with β=3. The distance parameter remains constant with changing values of pheromone level and optimum path, as it is the ratio of pheromone level to optimum path level. For Bus 1, the pheromone level is 11.11038 with more optimum path value is 3.7034. (Same for Bus 2, Bus 3 and Bus 4). But the distance value remains constant as 3. (See Table 7)

Table 7: *HBO parameter values*

Number of buses	Pheromone Level	Distance	Optimum Path Level
Bus 1	11.11038	3	3.7034
Bus 2	10.39115	2.99	3.4638
Bus 3	10.3147	3	3.4382
Bus 4	10.9445	3	3.6481

In Greedy Randomized Approach, the distance between two parameters is calculated based on Euclidian distance. The pheromone level and optimum path level are based on distance. If the distance between two stops is more, pheromone level and optimum path level gets decreased otherwise it's increased. Greedy Randomized approach considers the good result obtained for q0, β and p as 0.95,3 and 0.7. By considering the maximum value for pheromone level and optimum path level the distance travelled by each bus is also increased. (See Figure 4)

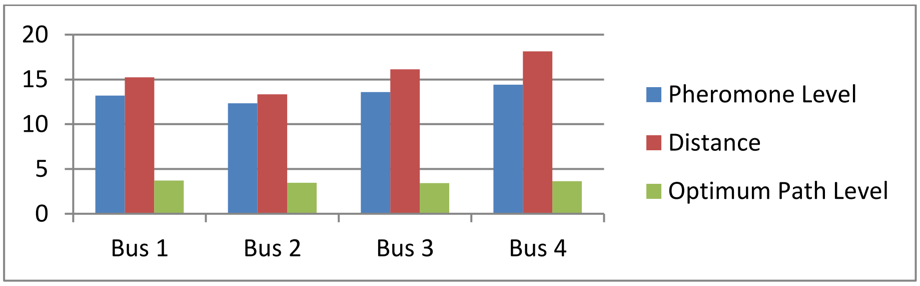

Figure 4: *Greedy Randomized Approach output with parameter values*

The pheromone level is increased for that it is calculated with q0=0.95 and the optimum path level increases, it is calculated with β=3. The distance parameter also changes with respect the values of q0, β and p=0.7. For Bus 1, the pheromone level is 13.1967 with more optimum path value is 3.7034 and the distance coverage is also more compared to ACO and HBO i.e. 15.2375 (Same for Bus 2, Bus 3 and Bus 4). (See Table 8)

Table 8: *Greedy Randomized Approach parameter values*

Number of buses	Pheromone Level	Distance	Optimum Path Level
Bus 1	13.1967	15.2375	3.7034
Bus 2	12.3399	13.3231	3.4638
Bus 3	13.57206	16.1174	3.4382
Bus 4	14.4007	18.1448	3.6481

As seen in previous section, the analysis of ACO, HBO and Greedy Randomized Approach is carried out based on pheromone level, distance and optimum path level. It is observed that the performance of all techniques is different in terms of output given by them for each parameter. From this we can say that instead of applying one technique to solve problem in hand, one can build the hybrid model to solve the problem to a great extent.

6 Conclusion

In this research work, we have implemented the approaches to solve a real-life school bus routing problem. One can solve it using a two-phase resolution approach. The first phase is to define the assignment of student pickup (or student delivery) points to buses, while the second phase is the actual routing algorithm. The problem is solved using one of the routing algorithms, which gives us the shortest route as well as optimal solution based on different predefined constraints. Also the major important task is we have done the analysis of ACO, HBO and Greedy randomized approach to decide which algorithm suits to the problem. But the observation says that none of them gives better performance for all parameters, instead all are good for different parameters.

References

[1] Li, L. and Fu, Z., “The school bus routing problem", Journal of the Operational Research Society,Department of Industrial and Management Engineering, Pohang University of Science and Technology (POSTECH), European Journal of Operational Research, 2009.

[2] Jorge Riera-Ledesma _, Juan-Jose´ Salazar-Gonza´ lez, “Solving school bus routing using the multiple vehicle traveling purchaser problem: A branch-and-cut approach”, Computer & Organization Reaserch, 2013.

[3] Jorge Riera-Ledesma _, Juan-Jose´ Salazar-Gonza´ lez, “A column generation approach for a school bus routing problem with resource constraints”, Computer & Organization Reaserch, 2012.

[4] Junhyuk Park, Hyunchul Tae, Byung-In Kim, “A post-improvement procedure for the mixed load school bus routing problem”, Department of Industrial and Management Engineering, Pohang University of Science and Technology (POSTECH), Pohang, European Journal of Operational Research, 2011.

[5] Patrick Schittekat, Joris Kinable, Kenneth Sörensen , Marc Sevaux , Frits Spieksma , Johan Springael, “A metaheuristic for the school bus routing problem with bus stop selection”, European Journal of Operational Research, 2013.

[6] Ocotlan Diaz-Parra, Jorge A. Ruiz-Vanoye, Angeles Buenabad-Arias, Felipe Cocon, “A Vertical Transfer Algorithm for the School Bus Routing Problem”, Department of Information Technology, Mexico,2012.

[7] Sayda Ben Sghaier, Najeh Ben Guedri, Rafaa Mraihi, “Solving School Bus Routing Problem with Genetic Algorithm”, High Institute of Transport and Logistics, 2013.

[8] V.Selvi, Dr.R.Umarani, “Comparative Analysis of Ant Colony and Particle Swarm Optimization Techniques”, International Journal of Computer Applications, Tamilnadu, India, 2010.

[9] A.E. Rizzoli, R. Montemanni, E. Lucibello, L.M. Gambardella, “Ant colony optimization for realworld vehicle routing problems: From theory to applications”, 2007.

[10] Vi Tran Ngoc Nha, Soufiene Djahel and John Murphy Lero “A Comparative Study of Vehicles Routing Algorithms for Route Planning in Smart Cities” , UCD School of Computer Science and Informatics, Ireland, 2012.