International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:10, No:2, 2016

Decision Support System for Solving Multi-Objective Routing Problem

Ismail El Gayar, Ossama Ismail, Yousri El Gamal

***Abstract*—**This paper presented a technique to solve one of the transportation problems that faces us in real life which is the Bus Scheduling Problem. Most of the countries using buses in schools, companies and traveling offices as an example to transfer multiple passengers from many places to specific place and vice versa. This transferring process can cost time and money, so we build a decision support system that can solve this problem. In this paper, a genetic algorithm with the shortest path technique is used to generate a competitive solution to other well-known techniques. It also presents a comparison between our solution and other solutions for this problem.

***Keywords*—**bus scheduling problem, decision support system, genetic algorithm, operation planning, shortest path, transportation.

I. Introduction

TRANSPORTATION problem is one of the important research fields this days specially. The vehicle routing and the bus scheduling problems are the most problems faced today in the transportation companies, since vehicles and especially buses are the main resources for providing services to their customers [2]. The bus scheduling problem is a complex part of the operational planning process on transport companies and the large size of the real problem has led to the development of a large number of models and techniques that are applied according to the complexity and the characteristics of each company. The bus scheduling problem deals with multiple objects in same time, causing a complexity to the problem. This complexity led to make the problem hard to be solved with an accepted solution. The problem objective is to minimize the transportation cost by 1) minimizing the number of buses 2) minimizing the distance of the trip 3) minimizing the time of the trip [5]. We can use any objective to represent the problem [11]. We can measure the time by many ways; one of the methods to measure the time is by measuring the longest trip time from the start point until return back to the depot. Or we can measure the longest trip time regarding the first passenger picked up until he arrive to the depot, also we can use the average total time of the trip for the buses and this may be the best way to measure our solution. [6]

There are many heuristic techniques used, like genetic algorithm [3], [4], tabu search [12], scatter search [1] and simulated annealing. The most famous solution technique is the genetic algorithm technique [7]-[10].

Sponsor and support are received by Arab Academy for Science & Technology.

Ismail El Gayar is with the Arab Academy for Science & Technology, Alexandria, EGYPT (e-mail: ismailelgayar@gmail.com).

II. Problem Statement

Given a graph with a multiple paths between N locations and a number of Passengers, Q needs to be transferred between these locations by using number of buses B.

- *N is the number of Nodes in the Map*
- *B is the number of Buses*
- *Q is the number of all the Passengers*
- C_B *is the number of the Capacity of each Bus*
- Q_B *is the number of the Passengers of each Bus*

The average time of all trips is generated and compared with our result which is represented by:

$$\text{The total average trip cost} = \frac{\sum_0^B \text{cost of each trip}}{B}$$

The objectives of the proposed techniques are:

- Minimize(tmax) where tmax = Maximum {t}
- Minimize(tb) OR Minimize(db)
- Minimize (b)

All these objectives to satisfy the business need the small difference that can save a lot to a company or school that transfer employees or students every day. So this is the really need to solve such a problem. In the problem proposed, we suppose that there is a specific function for each path to determine the cost of time, distance and the traffic. This problem can be solved easily to achieve the first objective, which is to minimize time required for each bus. The last objectives can be easily achieved by using one bus only for all the passengers. The difficulty part here is how to solve and achieve all the objectives together, so we used genetic algorithm to obtain an acceptable solution.

The problem has many constraints which can be expressed as:

1) All (N) must be visited (all the nodes must be visited)
2) Each (N) can visited by more than one (B) (Node can be visited by more than one bus)
3) Q <= B * C_B (Capacity of Passengers <= (Number of Buses * Capacity of each Bus))

III. Proposed Solutions

The Decision Support System (DSS) used is mainly designed to get a solution to solve the problem. The DSS can solve the problem faster and can get more than one solution to the problem

A. The Technical Implementation

The system is developed using java graphical user interface application based on object oriented programming

methodology (OOP) to be easily used with good user experience to the user. The decision support system is using some known algorithms to solve the problem; genetic algorithm, Dijkstra shortest path algorithm, sorting & searching techniques. The system makes the following processes sequentially to solve the problem. We can use the initial population parallel to generate many solutions as an initial solution for the problem.

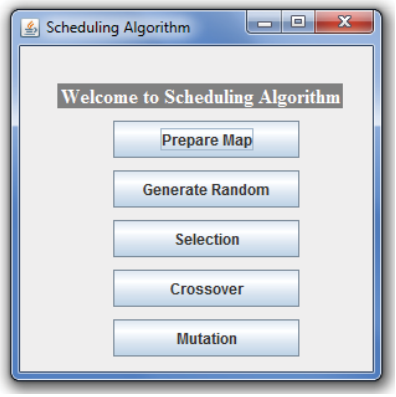

Fig. 1 The Application Interface

B.The Data Model Representation

The genetic algorithm depends on the chromosomes as a solution, we will make the genetic algorithm steps "selection, crossover, mutation" on it, in our solution we use the chromosome as a path as Fig. 2.

A	B	C	D	E	F	G	H	I	J
0	1	0	0	1	1	0	0	1	0

Fig. 2 The Chromosome Representation

This chromosome represents the Path "B, E, F, I"; we represent the solution as multiple chromosomes represented by a matrix containing the number of busses as rows and the number of the nodes as columns. So, in a solution which contains 5 buses and 10 nodes, we will have matrix of 5*10, we can model the Solution as a matrix in Fig. 3.

A	B	C	D	E	F	G	H	I	J
1	0	0	0	1	0	1	1	0	0
0	1	0	1	0	1	0	0	0	1
0	1	0	0	1	1	0	0	1	0
0	0	1	0	0	0	0	1	0	0
0	0	0	1	0	0	1	0	1	0

Fig. 3 The Solution Representation

C.The Algorithm Technique

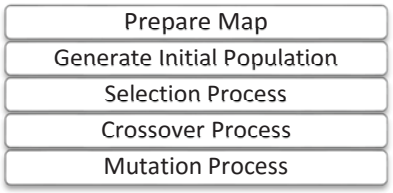

Fig. 4 The Process Steps

- "*Prepare Map*" is used to find the shortest cost between each node and the other nodes. The cost is evaluated using Djikstra shortest path algorithm which determines the lowest cost between the nodes.
- "*Generate Initial Population*" is to generate four solutions represented as a matrix using a function by creating a path using the shortest path to visit the nearest node using the first bus. Then we create another path to visit the remaining nearest node and so on until we find validated solution. We validate whether the solution is accepted and completed or if it cannot be a complete solution. This validation is to ensure that each node is visited at least one time and all passengers have been picked up. And also, we must ensure that each bus visits at least one node to ensure that all the buses picked up at least one person. After that, we generate other solutions after generating the first solution using the shifting bits technique. We use each path and shifting the matrices to the left to make the bus visit in the second solution "D" then "G" then "J" instead visiting node "C" then "F" then "I". It will be so that the route of the trip will be changed and the total cost of the solution also will be changed. We use the shifting left by one bit to find all the complete solutions and find the best three solutions of them as to be our generated second and third and fourth solutions.
- "*The Selection Process*" is to apply a specific function to evaluate the generated random solution to find the efficiency of the four solutions generated above "by evaluating the cost of each solution" to select the best two random solutions having the best cost using the fitness function to get the minimum total average trip cost for each solution.
- "*The Crossover Process*" is used for applying the crossover step on the two generated random solutions selected which have the best fitness function results. We are trying to make all the possible solutions by changing the rows using the both solutions. Then we use to evaluate the solution results from the crossover by the specific function used before to find the best solution we can use.
- "*The Mutation Process*" is used to fill the bus capacity in each node visited so we can make a balance to the solution. We need to mutate the cost to adjust the buses' capacity until we can make the solution balanced by enhancing the weights. After each enhancement we verify that all the passengers are served

IV. Result

Regarding to our experimental results, we tried to generate solutions for other system samples. And we got results which are not better to the models used before but they are other ways to solve the problem. This result can be improved by using other techniques hybrid to this technique. Fig 5 shows samples we tried with the results compared to other results from different techniques used by researchers before. We communicated with the techniques' owners and got the samples. then we test our solution compared to this alternatives samples [1]. According to this sample, we have the following results.

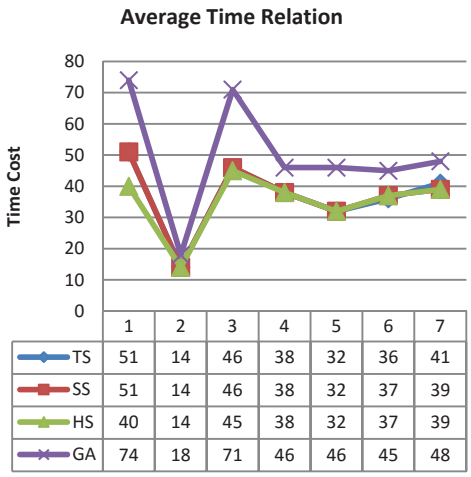

Fig. 5 Average Time Relation

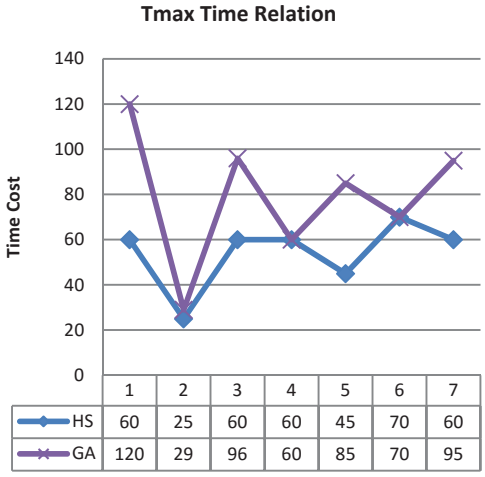

Fig. 6 Tmax Time Relation

TABLE I
Average Time Comparison

Sample	Buses	HS	GA	Percentage
1	4	40	**74**	**85%**
2	3	14	**18**	**28%**
3	5	45	**71**	**57%**
4	3	38	**46**	**21%**
5	6	32	**46**	**43%**
6	6	37	**45**	**21%**
7	5	39	**48**	**23%**

TABLE II
Maximum Time Comparison

Sample	Buses	HS	GA	Percentage
1	4	60	**120**	**100%**
2	3	25	**29**	**16%**
3	5	60	**96**	**60%**
4	3	60	**60**	**0%**
5	6	45	**85**	**88%**
6	6	70	**70**	**0%**
7	5	60	**95**	**58%**

The above results showed that the TS "Tabu Search", SS "Scatter Search" and HS "Heuristic Search" techniques and our technique the GA "Genetic Algorithm" technique are competitive techniques. As in the first sample the heuristic search is the best technique while from the second to the fifth, the Tabu and the heuristic techniques are the best, while in the sixth and seventh our technique performing the best. We can conclude that some of the Genetic Algorithm results are not efficient referred to Heuristic search because the heuristic search technique used the Tabu and Scatter Search techniques to find the best results of both algorithms then use the best result to be the final solution.

V. Conclusion

In this paper, we found that the bus scheduling problem is a complex problem which has no direct solution especially in the large networks. Interconnected problems require extensive knowledge of transit operations and legal labor agreements. The results proved that genetic algorithm could quickly produce satisfied results closer to the expectation, which can be enhanced to find better solutions. In our system, the system can generate solutions with a good performance with the medium size problems.

References

[1] Marti, "*Heuristic Solutions to the Problem of Routing School Buses with Multiple Objectives*", 2000.
[2] Spada, "*Decision-aid Methodology for the School Bus Routing and Scheduling Problem*", 2003.
[3] Kidwai, "*A Genetic Algorithm based Bus Scheduling Model for Transit Network*", 2005.
[4] Dias, "*A Genetic Algorithm for the Bus Driver Scheduling Problem*", 2001.
[5] Schittekat, "*A metaheuristic for solving large instances of the School Bus Routing Problem*", 2006.
[6] Schittekat, "*An efficient metaheuristic for the School Bus Routing Problem*", 2012.
[7] James, "*Decision Support System for Vehicle Scheduling in Uganda: A Case Study of Gateway Bus Company*", 2008.

[8] Bielli, "*Genetic Algorithms in bus network optimization*", 2002.
[9] Lourenco, "*Metaheuristics for The Bus Driver Scheduling Problem*", 2001.
[10] Suhl, "*Progress in solving large scale multi-depot multi-vehicle-type bus scheduling problems with integer programming*", 2008.
[11] Nayati, "*School Bus Routing and scheduling using GIS*", 2008.
[12] Shen, "*Tabu Search for Bus & Train Driver Scheduling with Time Windows*", 2001
[13] Torrance"*Vehicle and Driver Scheduling for Public Transit*", 2009.