Malaya Journal of Matematik, Vol. 9, No. 1, 71-78, 2021

https://doi.org/10.26637/MJM0901/0011

Evolutionary algorithm for the bi-objective green vehicle routing problem with time windows

V. Bugcy Mettilda[1]* and N. Meenal[2]

Abstract

The optimization of bi-objective vehicle routing problem has become a research hotspot in recent decades. In this paper, the bi-objective vehicle routing problem with time windows [BO-VRPTW] is proposed based on the existing research and a bi-objective mathematical model is formulated. This work gives focus on a bi-objective VRPTW to minimize both total distance and time balance of the routes. The main objective of this paper is to find the lowest –cost set of routes to deliver demand using identical vehicles with limited capacity to customers with fixed service time windows. This algorithm is applied for a publicly available set of benchmark instances, resulting in solutions which are better than others previously published.

Keywords

Vehicle Routing Problem with time windows, Bi-objective optimization, Evolutionary Algorithm, Genetic Algorithms.

AMS Subject Classification

90B06, 90C29, 68W50.

[1,2] *Department of Mathematics, JJ College of Arts and Science Autonomous, [Affiliated to Bharathidasan University, Tiruchirappalli-620024, India.] Pudukkottai-622422, Tamil Nadu, India.*

***Corresponding author**: [1] metti14v@gmail.com; [2]meenal_thillai@rediffmail.com

Article History: Received **13** October **2020**; Accepted **21** December **2020**

Contents

1. Introduction

The Vehicle Routing Problem (VRP) is a complex combinational optimization problem and has several variants of increased difficulty, in particular, the one with time windows (VRPTW) which has both capacity and time constraints.

The Vehicle Routing Problem with time windows (VRPTW) which was first introduced by Solomon in 1987[1]. In the VRPTW, each customer is assigned with a time window, which must be served within the time window. Optimal solutions for small instances of VRPTW can be obtained using exact methods. But the computation time required increases considerably for larger instances. This is the reason that many published studies have made use of heuristic methods. The recent surveys by Braysy and Gendrean (2005) [2,3] gives a complete list of studies about VRPTW and a comparison of the results have been made.

1.1 *V*RPTW as a Bi-Objective Problem

Over the years, several publications employing evolutionary algorithms have published to solve VRPTW as a single – objective optimization problem (Desrochers et.al 1992, Debet.al 2002, Berger et. al 2003, Gracia-Najera and Bullinaria 2008), [4, 5, 6, 12]. Recently Le Bouthillicar and Crainic (2005) [7] both have presented a parallel co-operative multi search approach for VRPTW which is based on the solution warehouse strategy in which several search threads co-operate by asynchronously exchanging information on the best solutions have found. These search methods implement a different meta-heuristic, an evolutionary algorithm or a tabu search procedure.

Homeberger and Gebring (2008) [8] presented a two-phase hybrid Meta–heuristic to find solution for VRPTW. This was the first phase aimed at the minimization of the number of routes by means of (μ, λ) evolution strategy, whereas the total distance is minimized in the second phase by using a tabu search procedure.

The bi-objective vehicle routing problem (BOVRP) is another extension of VRP. The classical way to solve BOVRP is to join or add the two objectives together and then solve it as a single –objective optimization problem. This is the method essentially to search for a single – objective optimal solution.

In the past few years a couple of studies have been presented that are of special relevance to us since they treated VRPTW as a bi-objective optimization problem, minimizing the number of vehicles and the total travel distance and used a genetic algorithm for solving it.

Ten et. al (2006) [9] put forward a hybrid multi – objective evolutionary algorithm [HMOEA] to deal with BOVRP, in which two objectives are number of vehicles and travel distance [9]. They used the dominance rank scheme to assign fitness to individuals and designed a crossover operator for the specific problem known as route – exchange crossover and applied a multi-mode mutation which is considered as swapping, splitting and merging of routes. Ombuki et. al (2006) [10] designed a multi – objective genetic algorithm to minimize the number of vehicles and travel distance. They also proposed the problem – specific genetic operators best cost route crossover and constrained route reversal mutation, which is an adaptation of the randomly used inversion method. Chiang and Hsu (2014) [11] proposed knowledge –based evolutionary algorithm for the solution of bi-objective vehicle routing problem with time windows which minimize the number of vehicles and distance.

In this paper, the presentation of work is concerned with the solution of VRPTW, as a bi-objective problem using an evolutionary algorithm (BiEA) which is applied in the genotype space to select parents for the recombination process. This leads to find the good solutions to the problem. The proposed algorithm has tested on publicly available benchmark instances and the results are compared with those from recent publications. Thus the algorithm appears very competitive.

The rest of this paper is organized as follows. In section2, VRPTW is introduced in more detail, the proposed BiEA for solving VRPTW as a bi-objective problem is explained in section3. In section4, a genetic algorithm for Bi-GVRP has explored. In section5, the results achieved by the algorithm have presented, as well as the comparison with some others which are already published. Finally conclusions and future work opportunities are added in section6.

2. The Vehicle routing Problem with Time Windows

The vehicle routing problem (VRP) is a complex combinational optimization problem and has several variants of increased difficulty, in particular, the one with time windows (VRPTW) which has both capacity and time constraints.

First, there is a need to specify the information involved in an instance of this problem before defining VRPTW. The VRPTW is more complex comparing with the travelling salesman problem (TSP) as it deals servicing customers with time windows using multiple vehicles (lawler et. al 1985) [13]. The optimal solutions to the VRPTW can be found using exact methods, the computational time required for solving VRPTW optimally is prohibited for large problem (cordeau et. al 2002) [14].Therefore heuristic methods are often used to solve optimal or nearly optimal solution in a reasonable amount of time. Heuristic approaches use route construction, route improvement or growth that combines both route construction and route improvement.

Recently, the problem in VRPTW is prolonged to the real-world business .Berger and Barkaoui (2004) [15] propose a parallel version of a hybrid genetic algorithm for VRPTW. This way of dealing is based on the simultaneous evolution of the two populations of solutions try to focus on separate objectives subject to secular constraint relaxation. Braysy and Gendreau (2005a, 2005b) [3] confer a survey of the research on the VRPTW. Both traditional heuristic route construction pattern or methods and recent local search algorithms are examined and given an overview of meta-heuristic approaches for the VRPTW.

Ghoseiri and Ghannadpour (2010) [16] constitute an evolutionary algorithm for the VRPTW by incorporating various heuristics for local exploitation in the evolutionary search and the notion of pareto's optimality. Garcia- Najera and Bullinaria (2011) [6] propose an improved multi- objective evolutionary algorithm for the VRPTW by collecting a similarity measure amidst solutions. All dealings or approaches are quite effective, as they confer solutions competitive with well –known benchmark data of Solomon's VRPTW instances (2008)[1].

2.1 Methodology

There is a need to specify the information involved in an instance of this problem .The model uses a fixed number of vehicles to serve a fixed number of customers from a single depot. Each customer has a known demand. They all have the identical capacity and have a delivery time window obtained by an earliest and latest arrival time. On the other hand the depot also has its time window. Time windows are soft in type for both customers and the depot. Here late arrivals lead to a penalty per time unit of tardiness and overtime cost takes into account.

2.2 Mathematical Notation

k: number of vehicles,
n: number of customers,
v_i: customer i, with index i=1…… n
d_i: demand of customer
s_i: service time of the node i
Q: capacity of each vehicle
C_{ij}: cost of vehicle between the nodes i and j
t_{ij}: travel time between customer i to customer j. where i,j=1…..n, i≠j
b_i: service time at customer i
e_i: earliest arrival time at customer i
D_k: Total demand for vehicle k.

First of all we have a set v={v_1…… v_n} of vehicles which is called customers. The customer u_i $\forall$ iε{1…..n} is located graphically at position (x_i,y_i) has demand of goods d_i>0 with time window [b_i,e_i] during which it has to be supplied and refers a service time s_i which is to unload goods. The vertex v_0 called the depot located at (x_0,y_0) with d_0=0 and time window [0,e_0= max{e_i:i∈{1…..n}}] from which customers are serviced utilizing a fleet of identical vehicles with capacity Q =max {d_i: i∈{1…..n}}

The travel between vertices v_i &v_j has an associated symmetric cost $c_{ij} = c_{ji}$ $\forall$ i,j ∈{0…..n}
which is usually considered to be the Euclidean distance. A vehicle may arrive early at the customer location but it has to wait until the beginning of the time window .Late arrival is not allowed .So it is commonly to have time t_{ij} to travel between vertices v_i and v_j just to be t_{ij}=c_{ij}.

Here the problem consists of designing a minimum cost set of routes and each route starts and ends at the depot. So that each customer is serviced by exactly one vehicle. Therefore each vehicle is assigned to a set of customers and that has to supply but at the same time the sum of their demands cannot exceed the vehicle capacity.

r_k=<$u_1{}^k$……..$u^k_{n_k}$ > is denoted as the k^{th} designed route and supplies n_k customers with $u_i{}^k$ the i^{th} vertex to visit in the route. Noting that in that notation the depot is omitted. But the depot is to consider before the first customer and after the last customer. Finally the customers demand D_k associated with route r_k is as follows.

$$D_k = \sum_{i=1}^{n_k} d^k_{u_i} = Q \tag{2.1}$$

.

In the same way the cost c_k associated with route r_k is defined as

$$c_k = c_{ou^k_1} + \sum_{i=1}^{n_{k-1}} c_{u_i}{}^k u^k_{i+1} + c_{u^k_{n_k}} 0. \tag{2.2}$$

Once defining the problem at least two objective functions are identified and that could be minimized.
If $R = \{r_1 \ldots . r_m\}$ is the set of designed routes, we can consider minimizing the number of routes

$$f_1(R) = |R| \tag{2.3}$$

and the total cost

$$f_2(R) = \sum_{k=1}^{|R|} c_k \tag{2.4}$$

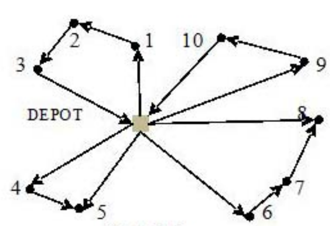
.

3. Bi-objective evolutionary approach for GVRTW

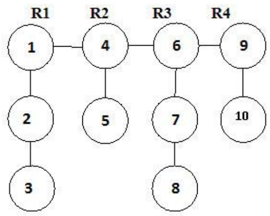
In this section the proposed EA for solving VRPTW is presented as a bi-objective problem. Here the encoding of the solution and the stages of processing involved are explained in detail.

3.1 Encoding for Solution

Here we use a tree representation in which every node has at most two children. The child which is in left represents the following customer to meet in a route. The right child points out to the route which is next in the solution. A solution to an example instance and its representation are figured.



(a) Solution



(b) Encoding

Fig1. Solution to an example instance of VRPTW and its encoding.

In the figure, the allocation of customers to routes, and the sequence will be serviced within each route, as follows:
customers 1,2 and 3 are to the first route,
customers 4 and 5 are to the second,
customers 6,7 and 8 are to the third,
customers 9 and 10 to the fourth route.

3.2 Fitness Assignment

When a single – Objective problem is solved using an evolutionary algorithm, Fitness assignment takes place to an individual according to its objective function evaluation. Whereas in the multi- objective case, Fitness cannot assigned straight forwardly, due to there being not only one objective function, but at least two of them. Therefore, in this work, the non- dominance sort criteria [4] is used to assign fitness to solutions, where the population is separated into several non –dominated fronts and depth identifies the fitness of the individuals belonging to them. In this case, the lower is the front and the filter is the solution.

3.3 Evolutionary Process

The algorithm begins with a set of feasible random solutions and each contains a set of randomly generated solutions. These routes are built as follows.

1. A customer is chosen and placed as the first location to visit on that particular route.

2. A second customer is selected. If capacity and constraints are to meet, then it is placed after the previous one.

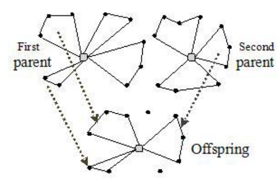
3. If none of the constraints are met, a new route is found and this customer will be first location to visit in the new route.

4. This process is repeated until all customers have assignment to a route.

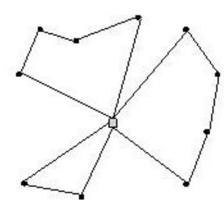
5. Now the objective functions take place to evaluate for every solution in the population and they are all got fitness assignment value.

3.4 The Recombination Process

The evolution continues with recombination of two parents are shown in the following figure 2(a). Here the algorithm is to aim at preserving routes from both parents. At first random number of routes are found from the first parent and are copied into the offspring.



(a) Copying routes from parents



(b) Offspring

Fig 2. The Recombination Process

Then all those routes from the second parent which are in conflict with the customers which is already copied from the first, are reproduced into the offspring. In this case, both routes on the left from the first parent were chosen to be replicated into the offspring and can only copy the route on the right from the second parent, as the other two contain customers, who already present in the offspring. If the unassigned customers remain, these are allocated, in the order they appear in the second parent, to the route where the lowest travel distance is attained as in figure 2(b).

Once an offspring has been generated, it is introduced to the mutation process. In this algorithm there are possible mutation operators, and can be placed as inter - and intra-route.

3.5 Mutation process

This algorithm will execute differences between two routes, thus the assignment of customers to routes is modified and then the differences will be done within a route and affect the travel sequence.
At First, we have to identify two variable processes which include:

1. Eliminating a sequence of customers from a route and inserting it into another.

2. Swapping two sequences of customers from various routes.

We use three operations under the case of intra-route.

1. The inversion of the sequence of a sub-route.

2. The shift of one customer.

3. Splitting a route.

Thus operations are shown graphically in Figure 3.

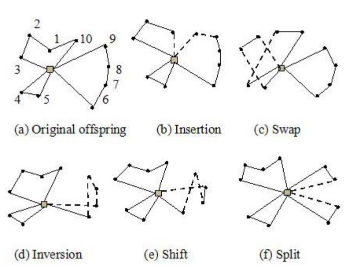


Fig.3. The Mutation Operators used in BiEA

The changes in the sequence are represented by dotted lines. As we see in figure 3(b), customer 10 was removed from the left route and has been inserted in the right route. The swapping of customer 4 with customer 2 and 3 are shown in figure 3(c). We have the inversion of the sequence of customers 7,8 and 9 in figure 3(d). The customer 9 has been moved or shifted between customers 6 and 7 as shown in figure 3(e). Finally the route on the right has been split between customers 7 and 8 which is figured in 3(f).
Noting that, all the mutation operators are not applied each time an offspring is mutated. First the split operator is effectuated with a probability equal to the inverse of the number of routes in the solution .Then the solution is obeyed to one of the inter – route operators. This decision is applied in random. Finally, any one of the intra – route operators is used to get the solution. The full mutation process is shown in following figure 4.

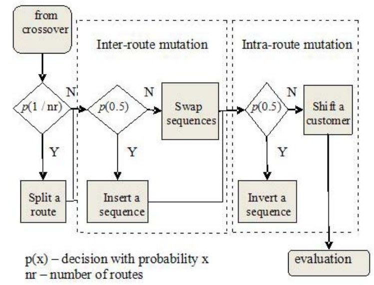


Fig 4. The Full Mutation Process

After the process is over, the algorithm starts to evaluate the objective functions for each solutions in the offspring population and merges both parent and offspring populations for fitness assignment. The solutions which are having the highest fitness are taken into next generation. If one front is contradicted with the population size, solidarity is calculated for such solutions in that front, and the less common are referred for the next iteration.

4. Genetic Algorithms for the BIGVRP

This work is an attempt to explore some of the potential of artificial evolutionary techniques (Goldberg 1989) [17]. The main goal is to maintain a population which is called a set of solutions (individuals), through a fixed number of iterations. A numerical value which means fitness depends on the function objective of the corresponding. Noting that, at each iteration, a number of individuals are chosen according to their fitness in order to form new ones by using two operators: crossover and mutation. A replacement is applied at the end of the iteration, to select the individuals which move to the next iteration.
The Genetic algorithm is structured as follows:

1. (a) Coding the solutions
   (b) Generating the initial population
   (c) Repeating
2. Selection
3. Crossover
4. Mutation

4.1 Chromosome Representation in GA

In a GA, each vehicle identifier denotes a separator in the chromosome between two alternate routes for solving Bi-GVRP, and a string of customer identifiers is represented by the sequence of deliveries that a vehicle should suffuse during its route. In Figure 5, the representation of possible solution for 2-customers and 4 vehicles is shown, noting that each route starts and ends at the depot.

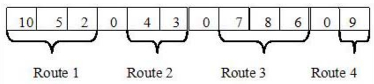


Fig 5: Typical representation of each chromosome

4.2 Initial Population:

First, an initial population is constructed such that each individual must be at least a feasible candidate solution, which means every route of the initial population should be feasible and randomly initialized.

4.3 Fitness Function and Selection:

The Fitness function of every chromosome (parent) in the population is assessed. It is found distance given. The higher fitness, t is like to be for the lower total distance. The computed fitness assists to choose members for the next generation.

The roulette type of selection is used in this work. A selection of roulette is an operator in which the chance of a chromosome selected is proportional to its rank (fitness). In this concept, the survival of fitness comes into play.

4.4 Crossover

The crossover which is a genetic operator that merges two parent chromosomes to generate a new offspring chromosome. The chromosome (Children) may be better than both of the parents, if it carries the best characteristics from each of the parent chromosomes in the idea being crossover operator. Crossover takes place during evolution due to a user definable probability of crossover.

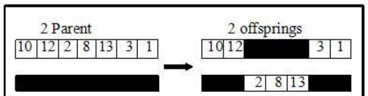


Fig 6: Figure 7: Example of 2-point crossover

4.5 Mutation

The “Mutation” is another genetic operator and is applied to a single solution with a certain probability. The small random changes are made by mutation operator. These random changes will slowly add some new significance to the population, which is not provided by the crossover. Here, partial-mapped crossover (PMX) and swap. Mutation [13] is applied for genetic operations of permutation based chromosomes. Further, the elitism.

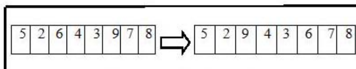


Fig 8: Example of Mutation

Strategy has a small number of good individuals and replaces the individuals which are worst in the next generation without going through the usual genetic operations.

4.6 Genetic Parameters

Genetic parameters consist of population size, number of generations, rate of crossover, and rate of mutation from existing works. In addition to that genetic parameters also include a operator named Selection which selects a chromosome randomly from the population. It is very through to find the best parametric values, so some of the following genetic parameters were applied through repeated experiments as shown in the table 1.

Genetic Parameters	**Applied Values**	**Ranges**
Population size	20	20-200
Number of end generations	600	500-20,000
Selection	Random	
Crossover Rate	0.6	0.50-1.0
Mutation Rate	0.01	0.003-0.01

Table - 1

5. Results and Discussion

Typically, to test the new algorithm, the well-known benchmarking problem is used due to Solomon [1]. Though several authors have published instances, the instances of VRP with Time windows which is published by Solomon are the uttermost largely used in experiments. The instances contain information per customer including its location, demand, time windows and service Duration. Location is expressed as (x,y) coordinate Time Windows expressed as an interval from an earliest time to Latest-time of start of service. The instances are categorized as clustered(C-type), Random (R-type). The 56 instances of Routing Problem with Time Windows (VRPTW) designed by Solomon in 1983[1] which contain 100 customers. But, later versions are available for 25 and 50 customers. The proposed algorithm is tested and experimented on Solomon’s VRPTW benchmarking problems with R1&R2, C1&C2, and RC1&RC2 which includes 56 instances. All 56 instances contain 25, 50 or 100 customer nodes and have a single depot.

The data are generated randomly in problems sets R1 and R2, clustered in problem sets C1 and C2, and a mixed structure (Random and Clustered) in RC1 and RC2 problem sets. There are differences between the problems R1 (C1, RC1) and R2 (C2, RC2). That is the instances in R1 (C1, RC1) have a shorter scheduling horizon and accept only a few customers per route, whereas the instances in R2 (C2,RC2) have a longer scheduling horizon allowing many customers to be served in that same route. The outcome (solutions) have been compared with the solutions using Manisri et al. (2009) [18].as Shown in Table1.

The results with comparison are split into two objective functions: One is the minimum number of vehicles and the other is minimum total travel times. NV represents the number of vehicles and TT represents the total travel time (on average).

Problem	**Result of GA**	**Number of Customers**			
		25	**50**	**100**	**All**
R1	NV	3.75	6.17	12.50	7.47
	TT	433.46	702.84	1326.92	821.07
R2	NV	2.0	4.09	7.27	4.45
	TT	470.54	702.69	1227.61	800.25
C1	NV	3.33	5.89	17.33	8.85
	TT	252.33	481.91	1563.79	766.01
C2	NV	1.88	3.13	12.38	5.80
	TT	279.88	520.31	1452.96	751.05
RC1	NV	3.25	6.75	13.50	7.83
	TT	354.28	711.19	1595.80	887.09
RC2	NV	2.63	4.75	8.25	5.21
	TT	432.14	739.89	463.30	878.44

Table - 2

The effectiveness of the results is illustrated by the proposed GA Algorithm and thus it provides competitive solutions with best solutions and also confers better solutions other than previously published.

6. Conclusions

In the present study, the proposed evolutionary Algorithm for the BO-VRPTW provides an effective solution and an efficient genetic algorithm was suggested for solving BI-GVRP formulation which incorporates the concept of pareto's optimality for the multi-objective optimization. In the proposed genetic algorithm, the number of vehicle and total travel distance is minimized simultaneously. We have compared the results from a single-objective genetic algorithm and with algorithms from recent publications by other authors. In the numerical experiment an instance in Solomon benchmark set is applied to test the performance of our algorithm. The objectives are to minimize both number of vehicles and time costs. Meanwhile two objectives are optimized separately and added together for making comparison for optimization. It is trivial that the results of bi-objective optimization are competitive. We are now looking at further ways to explore the extension of our approach to the minimization of at least one more objective which would be the waiting time. Finally we also have the plan to apply our B_iEA performance of different multi-objective optimization algorithms.

References

[1] Solomon, M.M: Algorithm for the vehicle routing and scheduling problems with time window constraints, *Oper. Res.,* 35(2)(1987), 254–260.

[2] Brasy, U., Gendreau, M.: vehicle Routing Problem with Time windows, part1: Route construction and local search Algorithm, *Transport. Sci.,* 39(1)(2005), 104–110.

[3] Brasy, 0., Gendreau, M.: Vehicle Routing Problem with Turie Windows, part II: Metaheuristics, *Transport. Sci.,* 39(1)(2005), 119–130.

[4] Desrochers, M., Desrosiers, J., Solomon, M.: A new optimization Algorithm for the vehicle Routing Problem with Time windows, *Oper. Res.,* 40(2)(1992), 342–350.

[5] Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and Elitist Multi objective Genetic Algorithm, *NSGA-II.IEEE T. Evolut. Comput*. 6(2)(2002), 182–195.

[6] Berger, J., Barkoui, M., Braysi, O.: A Route-directed Hybrid Genetic Approach for the Vehicle Routing Problem with Time Wndows, *INFOR*, 41(2003), 179–190.

[7] Le Bouthillier, A., Crainic, T.G.: A cooperative parallel Mathematics for Vehicle Routing Problem with Time Windows, *Comput. Oper. Res*., 32(2005), 1685–1700.

[8] Homberger, J., Gerbring, H.: A Two phase Hybrid Metaheuristic for vehicle Routing Problem with Time Windows, *Eur. J. Oper. Res*., 162(2005), 220–235.

[9] Tan, K.C., Chew, Y.H., Lee, L.H.: A hybrid Multi objective Evolutionary Algorithm for solving vehicle Routing Problem with Time Windows, *Comput. Optim. Appl*., 4(1)(2006), 115–150.

[10] Ombuki, B., Ross, B.J., Hanshar, F.: Multiobjective Genetic Algorithms for Vehicle Routing Problem with Time Windows, *Appl. Intell.,* 24(1)(2006), 17–25.

[11] Chiang, T.C., Hsu, W.H.: A knowledge–based evolutionary algorithm for the multiobjective Vehicle Routing Problem with Time Windows, *Compt. Oper. Res*., 45(2014), 25–35.

[12] Gracia-Najera, A., Bullinaria, J.A.: A Multi objective Density Restricted Genetic Algorithm for the vehicle Routing Problem with Time Widows, *In: 2008 UK Workshop on Computational Intelligence Leicester*, UK, (2008).

[13] Christofides, N., Vehicle Routing. In E.L, Lawler J.K., Lensta, A.H., Rinnooy Kan and schmoys, D.B.: *The Travelling Salesman Problem: a Guided Tour of Combinatorial Opt*., 12(1985), 1–10.

[14] Cordeau , J.F., Gendreau, M., Laporte., Potvin, J,Y., Semet, F.: A guide to vehicle routing heuristics, *Journal of Oper. Res Society*, 53(5)(2002), 512–520.

[15] Berger,J., Barkaoui, M.: A parallel hybrid genetic algorithm for the vehicle routing problem with time windows, *Comput. Oper. Res.,* 31(12)(2004), 2037–2050.

[16] Ghoseiri, K., Ghannadpour, S.F.: Multi-objective vehicle Routing Problem with Time windows using goal programming and Genetic Algorithm, *Applied Soft Computing*, 10(4)(2010), 1096–1105.

[17] Goldberg, D.E.: *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley, New York, 1989.

[18] Manisri, T., Mungwattana, A., Janssens, G.K: *Algorithm for Multi-Objective Vehicle Routing Problem with Time Windows*, 2009.

⋆⋆⋆⋆⋆⋆⋆⋆⋆
ISSN(P):2319 − 3786
Malaya Journal of Matematik
ISSN(O):2321 − 5666
⋆⋆⋆⋆⋆⋆⋆⋆⋆