Proceedings of the International MultiConference of Engineers and Computer Scientists 2012 Vol II,
IMECS 2012, March 14 - 16, 2012, Hong Kong

Pick-up Scheduling of Two-dimensional Loading in Vehicle Routing Problem by using GA

Yuxi Shen, Tomohiro Murata

***Abstract*--This study considers the application of a genetic algorithm (GA) to the basic vehicle routing problem with tow-dimensional loading constraints (2L-CVRP). Vehicle routing problem encompasses a whole class of complex optimization problems that target the derivation of minimum total cost routes for a number of resources (vehicles, etc.) located at a central depot in order to service efficiently a number of demand customers. Several practical issues in the industry involving both production and transportation decisions are modeled as VRP instances and the hard combinatorial problems in the strong sense (NP-hard). 2L-CVRP is a generalization of the Capacitated Vehicle Routing Problem, in which customer demand is formed by a set of rectangular, weighted items.**

In this paper, computational results are given for the pure GA which is put forward. As compare with the mathematic method, we can find the advantage of using GA to solve the problem especially in large size problems.

***Index Terms*— Vehicle routing problem; 2-dimesional Bin packing problem; Genetic algorithm; Capacity.**

I. Introduction

The 2L-CVRP is a particularly important problem. Its importance can be attributed to the fact that it is an interesting problem both from the theoretical and the practical points of view. Regarding the theoretical viewpoint, since 2L-CVRP is, in a sense, composed of two NP-hard optimization problems (CVRP and 2L- BPP), it is also a challenging NP-hard problem of high complexity. As far as its practical importance is concerned, the 2L-CVRP has an obvious commercial value. [1]

To the best of our knowledge, only two algorithm methodologies have been proposed for the 2L-CVRP. L. Caccetta, S.P. Hill [2] has developed an exact methodology which uses a branch-and-cut algorithm to deal with the routing characteristics of the problem and a branch-and-bound procedure to guarantee feasible loadings of the items into the vehicles.

Manuscript received May, 2011; revised Jan , 2012.

Yuxi Shen is a member of master laboratory with The Graduate School of Information, Production and Systems, Waseda University. Fukuoka 808-0135 Japan, (e-mail: sunkissmoon@fuji.waseda.jp)

Dr.Tomohiro Murata is a professor with The Graduate School of Information, Production and Systems, Waseda University. Fukuoka 808-0135 Japan, (e-mail: t-murata@waseda.jp)

This exact solution methodology is applied to problem instances with no more than 30 customers and 90 items. In practical conditions, the scale of the problem tends to be larger.

A variety of real life applications in the distribution or collection management context involves the transportation of rectangular shaped items that cannot be stacked, due to their weight or fragility (household appliances, delicate pieces of furniture, artworks, etc.).[3] So it can be widely used in manufacturing and service operations management applications:

--Pick-up schedule in logistic problems

--Minimization of the distribution costs in a multi-facility production system

--Routing problems for automated pick and place machines

--Yard trailer routing problem at a maritime container terminal

We find the traditional heuristic algorithm; such as MIP will cost very long time to solve the NP-hard problem, but the search mechanism in the GA corresponds to chromosome evolution, comprising reproduction, crossover, and mutation during imitated breeding process. [4] Typically, determining search direction solely relies on probability settings regarding chromosome mutation and crossover for generating offspring during the breeding process. Additionally, as is well know, the structure of the initial chromosome population (namely, initial solution population) significantly impacts the resolution capability of the GA. So we developed a new method by using GA that can effectively shorten the operation time.

II. Problem description

With paper surveys, we have studied two kind of placement version: The Unrestricted loading and the Sequential loading. [5] For the Sequential loading there is an additional constraint: the loading of the items must ensure that whenever a customer i is visited, no item of customer j which visited after customer i , can be placed between items of customer i and the rear part (loading door) of the same vehicle. [6] The sequence constraint arises in practice, when it is not feasible to move items inside the vehicle, due to their weight or fragility. The problem can be defined as follow:

Definite parameters

- i : The number of depots $(i = 1,\ldots,n)$
- D_{ij} : Distance between depot i and j where $i, j \in N$

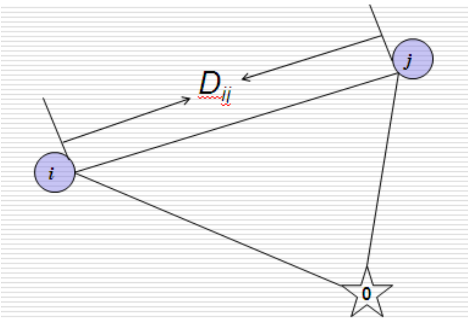

Fig.1. Problem definition.

- D: The weight capacity of each vehicle
- D_t: The total weight of each vehicle in the meantime.

$D_t = D_t + d_i$

- K: The number of vehicles
  - If ($D_t \le D$ *and* $W_t \le W$, $H_t \le H$) ; K++
- A : The volume capacity of each vehicle

1) All the items accessible from a single side for the rectangular loading surface ;

2) And width and height are equal to W and H ,

A = WH is the total area of the loading surface

3) The total width and height of items in each vehicle in the meantime : $W_t = W_t + w_{il}$, $H_t = H_t + h_{il}$

- Each item will be denoted by a pair of indices *(i,l)* .

1) Each depot i *(* $i = 1,\dots,n$*)* is associated with a set of m_i rectangular items whose total weight is equal to d_i ;
2) Each item has specific width and height equal to w_{il} and h_{il} *(* $l = 1,\dots, m_i$ *)*.
3) The total width and height of items in each vehicle in the meantime : $W_t = W_t + w_{il}$, $H_t = H_t + h_{il}$

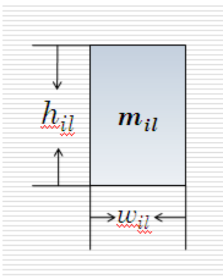

Fig.2. Define the parameters of each item.

Objective Function:

$$\min C = \left(\sum_{i=0}^{N} \sum_{j=o,i\neq j}^{N} \sum_{k=1}^{V} d_i^k x_{ijk}\right) \bullet C_s + K \bullet C_v$$

(C_v express the cost of each vehicle.)

And there is some special constrains for the problem:

(1) $x_{ijk} = 1$, if vehicle k travels directly from customer i to customer j (i, j ϵ N); 0 otherwise

(2) $d_i < D$ (i = 1,…,n)

(3) $0 < x_{il} < W - w_{il}$ and $0 < y_{il} < H - h_{il}$ (l = 1,…, m_i).

(4) $x_{il}+w_{il}<x_{jl'}$ or $x_{il'}+w_{il'}<x_{jl}$ or $y_{il}+h_{il}<y_{jl'}$ or $y_{il'}+h_{il'}<y_{jl}$

III. Solving Method

Step 1: Consider the condition of one vehicle with no capacity and time limited.

So the problem become very similar as the travelling salesman problem (TSP) [7] that was first formulated as a mathematical problem in 1930 and is one of the most intensively studied problems in optimization.

1) Formulate GAs for TSPs:

The evaluation function for the two dimensional TSP is the sum of Euclidean distances between each two cities in the problem. The fitness value can be defined as this:

$$Fitness = \sum_{i=1}^{i=N} \sqrt{(a_i - a_{i-1})^2 + (b_i - b_{i-1})^2}$$

Where a_i , b_i are the coordinates of city *i*. [8]

2) Crossover and mutation:

We considered a condition showed as Fig.3:

1	2	3	|4	5	6	P_1
2	6	5	|3	1	4	P_2
After		*Crossover*				
1	2	3	|3	1	4	C_1
2	6	5	|4	5	6	C_2

Fig.3. An example of the traditional crossover.

After crossover, we noticed that city 3 appeared twice in child 1 and the same as city 6. So the traditional crossover principle is not suitable for this case. Then I decided to use greedy crossover that invented by Mr.Grefenstette in 1985. [9] This method selects the 1st city of a parent, compares the cities leaving that city in both parents, and chooses the closer one to extend the route. If one city has already appeared in the route, we choose the other city. If both cities have already appeared, we randomly select a non-selected city.

For the same reason we do not use the traditional mutation method. We randomly select two cities in a chromosome and change their values. Then, we still have legal routes after the mutation.

3) Selection:

We considered two kind of search algorithm of selection. First one is the roulette-wheel selection that the proportion of the wheel is assigned to each of the possible selection based on their fitness value.

The other is chc selection which developed by L.J. Eshelman in 1991. [10] We use this selection to ensure the best one always survives in the next generation.

Fortunately, Sushi J Louis and Gong Li did an experiment to compare these two selection method. [11]

ISBN: 978-988-19251-9-0
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

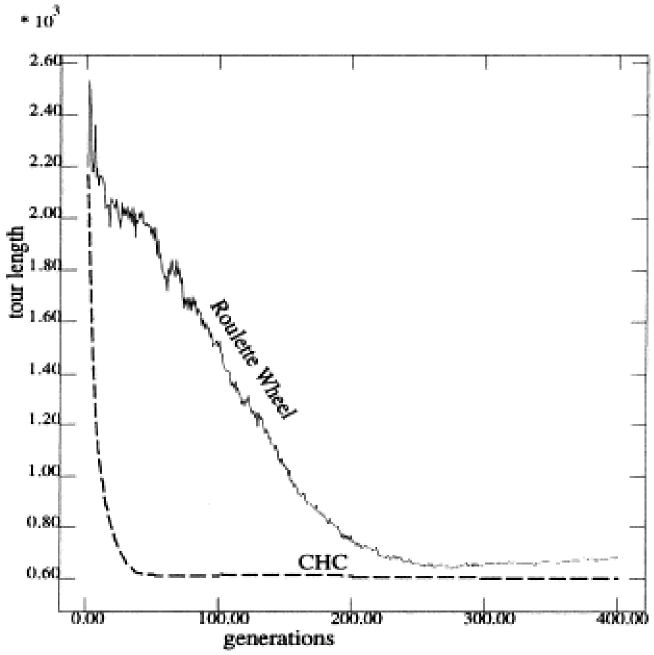

Fig.4. Compare CHC with roulette-wheel selection.

We can see how fast the chc selection did in the population converge from Fig.4. [12] The performance of chc selection is much better than the roulette-wheel selection in this case. So we decided to use chc for our selection.

4) Solutions:

Two parents are selected from population by binary tournament method. And, these two parents are chosen randomly. Then we calculate the fitness value, the one with the better fitness value is selected as the 1st parent we need. The process is repeated to obtain a second parent.
Children are produced by the two parent using the crossover method that we mentioned as above.
The steps of the pure GA can be described as follows:

Generate the initial population
Evaluate the fitness value of every individual in the population
Repeat
  Select two parents from the population randomly by using two binary tournaments
  Produce two children from the selected parents
  Do crossover and mutation
  Evaluate the fitness of the new offspring
  If entry criteria are satisfied by chosen offspring
    Choose population member to be replaced
  End if
Until stopping criterion is satisfied

So the main purpose here is to shorten the total distance that the vehicle travelled, because the cost is all produced by the distance.

Step 2: Add a weight and space capacity to the vehicle, and consider the load and unload in no time limited condition:

Here, the problem can be described as, one vehicle started from the warehouse, then visited several customers and pick-up their items. (Fig.5)When either the weight or space capacity is reached, the vehicle returns to the start point and unload all the items. Then restarted and continue to pick-up other customers' items, until all the customers' items are gathered to the warehouse.

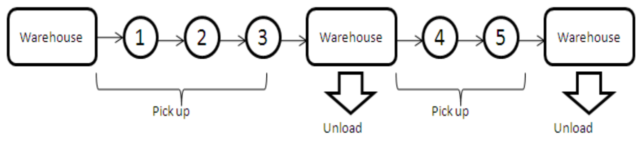

Fig.5. One vehicle with unload conditions.

So we need to consider the replacement problem here. As how to make good use of each vehicle's space to decrease the circle that the vehicle travels in order to save the cost.

In order to effectively reduce the number of possible packing methods, we decided to use a traditional placement heuristic called bottom-left-condition (BL algorithm). The orthogonal packing pattern fulfills the BL algorithm if no rectangle can be shifted further to the bottom or to the left.

Alternatively, a packing pattern can be represented by a permutation π .

i_j - Index of the rectangle ($r_{i_j}|$).

$\pi = (i_1 \ldots\ldots i_n|,)$ - Permutation.

This decoding of the genotype needs more effort than the conversion of the natural representation into the packing patterns. So the aim is to create a faster decoding algorithm.

Procedure 1: Place $r_{\pi(1)}$ into the left lower corner of the board.

Procedure i: Shift $r_{\pi(1)}$ alternately, beginning from the upper right corner of the board, as far as possible to the bottom and then as far as possible to the left.

For the genetic algorithm, an evaluation of the packing pattern is necessary. This is represented by an appropriate fitness-function

F: π **---> R+**
with the property
$f(\pi_i) > f(\pi_j)$

if π_i is a 'better' packing pattern than π_j .The computation of the natural approach of the fitness-function is inversely proportional to the height of the packing pattern:

$f(\pi) = 1/h_{BL}\ ('\pi)$.

If two packing method have the same height and fitness-values. There also has a condition that one of them is better than the other. (Fig.6)

For this reason a differentiated approach is necessary. In order to find a differentiated fitness-function the biggest resulting contiguous remainder among the packing patterns on the given board must be considered.

ISBN: 978-988-19251-9-0
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

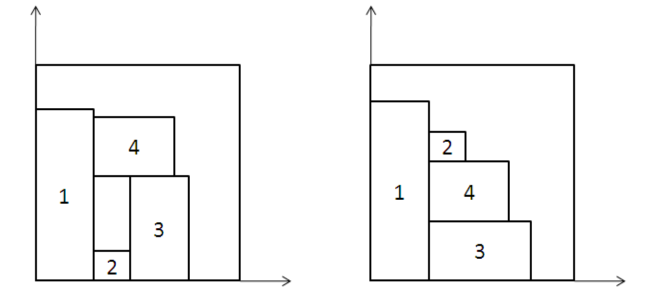

Fig.6. Two packing method with the same height and fitness-values that one is better obviously.

The comparison suggests the following fitness-function;

$$f(\pi) = \text{Area (Contiguous Remainder } (\pi))$$
$$= \sum_{k_i} \left| x_1^{(i)} - x_2^{(i)} \right| \bullet (height(board) - y^{(i)})$$

With

$$k_i = \{(x_1^{(i)}, y^{(i)}), (x_2^{(i)}, y^{(i)})\}.$$

Step 3: Muti-vehicles with time limit:

Now, we will give the last constraints to solve the reality problem. [13] Here we have a fleet of vehicles, each have the same weight and space capacity, and also have the same speed. (We didn't consider the effect to the speed that how many items have been loaded on the vehicles). [14]

Additionally, we add a time-limit for each case, so that each vehicle travels with its own circle and limited travelling distance. [15] [16] (we suppose there always have enough vehicles in the warehouse).

As the imagination showed in Fig.7, our decision parameter became the vehicle number we used and the distance.

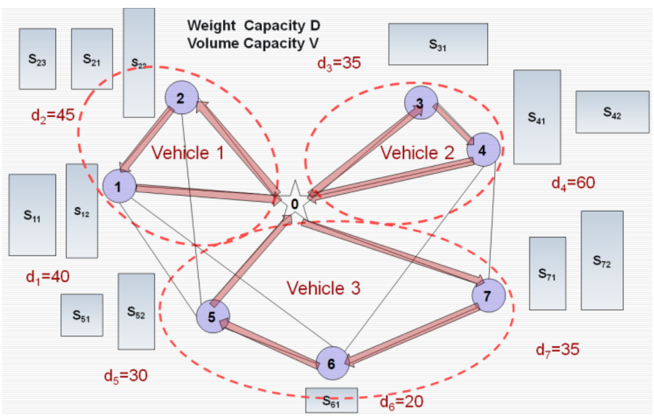

Fig.7. Muti-vehicles with capacity and time limit.

IV. Experiment and Evaluation

The heuristics defined above were coded in java and, using a Core(TM)2 Duo CPU E7200@ 2.53GHz x 2 with 2 GB RAM in Windows XP. They were applied to the CVRP problems. In a word, these experiments reveal relatively slight differences, and the GA remained effective with each of the variants that were tried.

The benchmark instances used in this paper and other papers are available at (http://www.or.deis.unibo.it/research.html). There are 36 instances and these instances are divided into five classes, by the number and the size of items:

We can see the result showed in TABLE I, we did 10 independent runs by increasing the customer and item number by using GA.

TABLE I
The Computation Result by Using GA.

Customers	Items	Vehicle hired (Avg)	Average Cost	Computing time(s)
5	12	2	543.92	8
10	23	2	906.13	12
15	31	3	1138.03	37
20	52	5	1831.45	89
30	77	7	3290.25	213
40	103	12	5601.21	470
50	132	15	8013.34	532
75	158	18	12281.32	1180
100	273	28	15020.12	2470

Then we used LINGO to solve the MIP (mixed integer programming) model and got the result of the total cost and computing time (TABLE II).

TABLE II
The Computation Result Ran by MIP.

Customers	Items	Vehicle hired	Total Cost	Computing Time(s)
5	12	2	543.92	21
10	23	2	893.01	80
15	31	3	1097.72	336
20	52	5	1764.65	1570
30	77	7	3098.67	5938
40	103	11	5132.73	18500
50	132	15	7714.22	33563
75	158	17	11253.12	75986
100	273	26	13460.94	196511

From comparing with the optimal solution, we calculate the Cost △% by

Cost△%=(Average Cost by GA －Optimal cost)/ Optimal cost;

to show how many percentage the cost will increase when we use GA, and we calculate the Computing Time△% by

Computing Time△%= Computing Time(GA)/ Computing Time(MIP);

TABLE III
Comparison of Two Methods

Customers	Items	Comparison of Vehicles	Cost△%	Computing Time△%
5	12	0	0	38.10
10	23	0	1.47	15.00
15	31	0	3.67	11.01
20	52	0	4.19	5.67
30	77	0	5.28	3.59
40	103	+1	9.13	2.54
50	132	0	5.88	1.59
75	158	+1	9.74	1.55
100	273	+2	11.58	1.26
			Avg:5.56	Avg:8.92

We can see the data from TABLE III that the cost that we will take by using GA is only 4.8% more than the best solution, and the computing time only take 7.81% in average compare with the mathematic method. In addition, when the customer number increased to 20 and the item number grown to 52, the computing time will have a great distinction by

ISBN: 978-988-19251-9-0
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

using these two methods, it reduced rapidly from 11.22% to 4.92%. And if we calculate the computing time △% from 20 customers to 100, the result is only 3.23% in average.(Fig.10)

But we find the result is not good enough when the customer number reached 40, 75 and 100, Cost△% increased to nearly 10%. That will cost a lot in the reality conditions. So we were thinking about if we can find a way to make some change of our chromosome design or solve method, that can reduce Cost△% lower than 5%.

With paper surveys, we find two-point crossover may work well than one-point crossover in some conditions. In the Two-point crossover, the ranking replacement method was replaced by a worst fitness/unfitness replacement method, in which the population member with the worst unfitness was chosen for replacement, unless all population members had zero unfitness, in which case the member with worst fitness was chosen for replacement. The child solution replaced the chosen member even if it was worse, as this caused a much larger number of child solutions to enter the population and gave rise to a better performance of the GA compared with only allowing the child to enter if it was better than the population member chosen for replacement as we used One-point crossover in the preceding experiment of the text.

Here we got the result by GA using the Two-point crossover. (TABLE IV)

TABLE IV
RESULT OF GA BY USING TWO-POINT CROSSOVER

Customers	Items	Vehicle hired (Avg)	Average Cost	Computing time(s)
5	12	2	543.92	7
10	23	2	911.08	12
15	31	3	1129.01	39
20	52	5	1829.37	66
30	77	7	3233.51	197
40	103	11	5408.01	452
50	132	15	8138.11	594
75	158	17	11968.57	993
100	273	27	14607.03	2169

Meanwhile, Cost△% also reduced within the decrease of vehicle number. The average of Cost△% down to 4.31% from 5.56%(TABLE V). In addition, the Computing Time △% almost remains the same(reduced from 8.92% to 8.23%,, that reveals after the experiment by using two-point crossover, GA remains efficient in the computing time comparison. We can get the result more closely with the mathematic method and we can always find the best vehicle number for the final solution until 75 customers. Actually, 75 customers' problem is large enough to serve the reality trade problems.

TABLE V
COMPARISON WITH MIP BY USING TWO-POINT CROSSOVER

Customers	Items	Comparison of Vehicles	Cost△%	Computing Time△%
5	12	0	0	33.33
10	23	0	1.98	15.00
15	31	0	2.85	11.61
20	52	0	3.67	4.20
30	77	0	4.35	3.32
40	103	0	5.36	2.44
50	132	0	5.49	1.77
75	158	0	6.36	1.31
100	273	+1	8.71	1.10
			Avg:4.31	Avg: 8.23

We can compare the figures in the two tables; the cost of these two solving methods appears very close when the customer number is small. And when the customer and items' number is increased, the total cost that calculated by the mathematic model can get the best solution for sure, but the computing time is much longer than GA. (Fig.8)

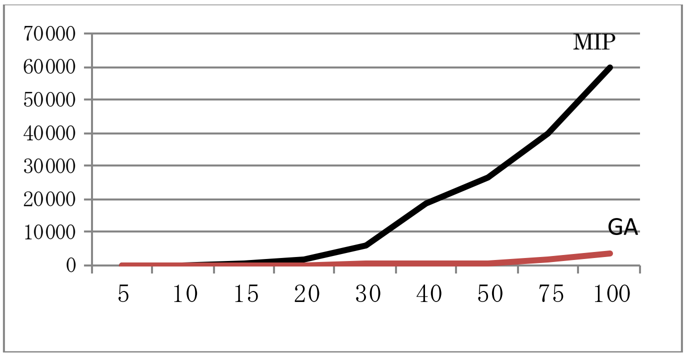

Fig.8. Contrast of computing time between GA and MIP.

And the average cost we calculated by GA is almost the same when contrasted with the best solution when customer and item number is low. Even the customers' number increased, GA only cost 5% more than the optimal solution. It will be acceptable in most reality conditions.

V. CONCLUSIONS

In this research a combination of Bin Packing Problem and Vehicle Routing Problem in distribution logistics is considered, known as the two-dimensional loading vehicle routing problem. In the field of combinatorial optimization, loading and routing problems have been studied intensively but separately.

A meta-algorithm of GA has been discussed here performed well, although it does not equal the mathematic model that ran by MIP in terms of solution quality, but it only takes 8.92% in average of the computing time when we make a tradeoff between these two algorithms.

Then we improved our method for changing Two-point crossover instead of One-point crossover. Finally we were able to get an acceptable result that can accurately calculate the vehicle numbers up to 75 customers and only consume 4.31% cost in average and greatly reduced the computing time(8.23%). In this sense, genetic algorithm appeared very high-efficiency in solving the NP-hard problem.

REFERENCES

[1] Asvin Goel a,b,, Volker Gruhn b A General Vehicle Routing Problem European Journal of Operational Research 191 (2008) 650–660.
[2] Caccetta, S.P. Hill Mathematical and Computer Modeling, Volume 33, Issues 4-5, February-March 2001, Pages 517-532
[3] Chi-Guhn Lee, Marina A. Epelman b, Chelsea C. White III c, Yavuz A. Bozer b A shortest path approach to the multiple-vehiclerouting problem with split pick-ups Transportation Research Part B 40 (2006) 265–284
[4] Guenther Fuellerera, Karl F. Doernera, Richard F. Hartla, Manuel Iorib,Ant colony optimization for the two-dimensional loadingvehicle routing problem. Computers & Operations Research 36 (2009) 655 – 673.
[5] Stephen C.H. Leung, Xiyue Zhou, Defu Zhang, Jiemin Zheng Extended guided tabu search and a new packing algorithm for the two-dimensional loading vehicle routing problem. Computers & Operations Research, Volume 38, Issue 1, January 2011, Pages 205-215
[6] Manuel Iori y, Juan Jose Salazar Gonzalez z,Daniele Vigo y An exact approach for the vehicle routing problem with two-dimensional loading constraints December 7, 2004; Revised July 1, 2005.

ISBN: 978-988-19251-9-0
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

[7] R. Radharamanan, L.I. Choi Computers & Industrial Engineering, Volume 11, Issues 1-4, 1986, Pages 236-240
[8] Andreas Bortfeldt A genetic algorithm for the two-dimensional strippacking problem with rectangular pieces European Journal of Operational Research 172 (2006) 814–837
[9] Ravindra K. Ahuja, James B. Orlin, Ashish Tiwari A greedy genetic algorithm for the quadratic assignment problem Computers & Operations Research, Volume 27, Issue 10, September 2000, Pages 917-934.
[10] Topon Kumar Paul, Hitoshi Iba Gene selection for classification of cancers using probabilistic model building genetic algorithm Biosystems, Volume 82, Issue 3, December 2005, Pages 208-225
[11] D. Greiner, G. Winter, J.M. Emperador Optimising frame structures by different strategies of genetic algorithms Finite Elements in Analysis and Design, Volume 37, Issue 5, May 2001, Pages 381-402
[12] Stefan Jakobs On genetic algorithms for the packing of polygons European Journal of Operational Research 88 (1996) 165-181.
[13] Iori M, Gonzalez JJS, Vigo D. An exact approach for the vehicle routing problem with two-dimensional loading constraints. Transportation Science 2007; 41(2):253–64.
[14] Nagy G,SalhiS. Heuristic algorithms for single and multiple depot vehicle routing problems with pick-ups and deliveries. European Journal of Opera-tonal Research 2005; 162(1):126–41.
[15] Enrique Alba, Bernabé Dorronsoro Computing nine new best-so-far solutions for Capacitated VRP with a cellular Genetic Algorithm. Information Processing Letters 98 (2006) 225–230.
[16] C.D. Tarantilis , G. Ioannou, G. Prastacos Advanced vehicle routing algorithms for complex operations management problems Journal of Food Engineering 70 (2005) 455–471

ISBN: 978-988-19251-9-0
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)