# HYBRID GENETIC-ALGORITHM / BRANCH & BOUND TECHNIQUE TO SOLVE A TIME-DEPENDENT TRANSPORTATION PROBLEM

**Octav Brudaru[1,2], Brent Valmar[1]**

[1]Technical University "Gh Asachi" Iasi, Department of System Engineering,
700050 Iasi, D. Mangeron 53, Romania
[2]Institute of Computer Science, Romanian Academy Iasi Branch,
700505 Iasi, Carol I 22A, Romania

*brudaru@tuiasi.ro (Octav Brudaru)*

**Abstract**

This paper deals with the solving of transportation problem in which a vehicle moves along a hamiltonian path with a decreasing charge for supplying the demands of a given set of consumers. The transportation cost is an affine function depending of the carried charge. The order to reach the consumers that minimizes the transportation cost must be found. This problem is NP-hard. It is proposed a heuristic method that combines a genetic algorithm with embryonic chromosomes with a branch and bound technique. In this way, the evolving mechanism operates with subsets of solutions rather than individual ones. The hybrid genetic algorithm explores a population of embryonic and adult chromosomes. The exploration of the state space tree organizations of solutions, that is specific to branch & bound, interleaves with the exploring of total solutions as a standard genetic algorithm does. A specific growing operator acts with a given probability in order to transform the embryonic chromosomes into adult ones. When geographic coordinates of the locations are available, the mutation and growing operators act on vicinities of the nodes. The size of these vicinities is smaller as the evolution advances and an effect similar to that of simulated annealing is induced. When the fitness function is applied to an embryonic chromosome it behaves like the estimate function used by the branch & bound techniques with the least cost strategy. A mechanism able to capture the intrinsic clustering information existing in the set of consumers is used in order to take benefit of the phase transition effect. The performance of the hybrid genetic algorithm regarding the solutions quality and the required computing time is experimentally investigated and reported.

**Keywords: genetic algorithm, branch & bound, transportation problem, hamiltonian circuit.**

## Presenting Author's biography

Octav Brudaru is professor of computing at Technical University ''Gh. Asachi'' Iasi, and a researcher at Institute of Computer Science, Romanian Academy Iasi Branch. His scientific interests and research activity include soft computing techniques, distributed genetic algorithms for combinatorial optimization, methods and tools for data clustering and approximation, design of hyperheuristics, operation management and the design of systolic algorithms.

ISBN 978-3-901608-32-2  

## Introduction

This paper addresses a transportation problem in which a vehicle has to supply a set of consumers with known demands. It starts from a unique warehouse charged with the whole demand, passes exactly once through each client delivering the corresponding demand and returns at the starting point. The transportation cost between two locations depends on the unitary cost between them and the weight of the transported charge plus the weight of the empty vehicle. The problem is to find a hamiltonian circuit that minimizes the total transportation cost. This problem generalizes the well-known travelling salesman (TSP) and it is on its turn NP-hard. Since the cost of traversing and edge $e = (i, j)$ depends on its position in the path this problems will be called time-dependent transportation problem (TDTP). The TDTP presents its own interest but it is contained as a critical subproblem in different variants of the vehicle routing problem [5]. In the second case, the fast and accurate solving of this problem is very important especially when it is invoked many times for computing the fitness function a genetic algorithm that is used for solving the main problem.

Some previous methods based on pure and hybrid genetic algorithms are presented in [2], [3], [4]. Some successful graftings of a branch at bound method on a genetic algorithm are reported in [2], [6], [7] and include the solving of difficult combinatorial problems. This paper presents a new combination of a genetic algorithm with a branch & bound technique for TDTP. This heuristic combines the best features of each of the components and its performance is better than those of the previous methods with respect to both solution quality and computing time. This new hybrid metohd uses embrionic representation and this ensures a better exploration of the search space at the beginning of the evolution when a large set of incipient solutions can be hosted. The fitness of the each individual combines the cost induced by the embrionic chromosome with the score given for the unknown part of the chromosome by the estimate function used by the branch & bound technique.

Section 2 contains the statement of the problem. In section 3, an outline of the branch & bound method which is grafted on the genetic algorithm and the way to compute lower bound for the transportation cost are presented. The main components of the hybrid genetic algorithm with embrionic chromosomes are described in section 4. The results of the experimental investigation done for tuning the parameters and estimating the performances of the hybrid genetic algorithm are presented in section 5. Last section summarizes the work.

### 2. Statement of the problem

Consider the digraph $G = (V, A)$, where $V = \{0,1,...,n\}$ is the set of vertices and $A$ is the set of arcs. The vertices $1,...,n$ represent the consumers and " 0 "denotes the depot. The integer $d(i)$ is the demand of consumer $i$, whereas $d(0)$ is the weight of the empty vehicle. The quantity $D = d(0) + ... + d(n)$ is the total charge.
For each $(i, j) \in A$, the positive integer $\mathrm{cost}(i, j)$ is the unitary cost of transportation from $i$ to $j$. The cost of transporting charge $r$ (including $d(0)$) from $i$ to $j$ is $c(r; i, j) = r * \mathrm{cost}(i, j)$.
Let $x = [0, x_1,..., x_n, 0]$ the hamiltonian circuit starting from 0, passing through $x_1,..., x_n$ and ending at 0. The cost of this circuit is

$$C(x) = c(D; 0, x_1) +$$

$$+ \sum_{i=1}^{n-1} c(D - \sum_{h=1}^{i} d(h); x_i, x_{i+1}) + c(d_0; x_n, 0).$$

The delivering activity along a tour is illustrated in Fig.1.

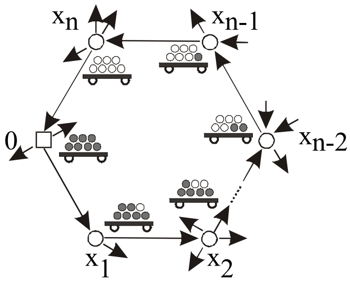


Fig. 1 Delivering along a hamiltonian circuit

If $H$ is the set of hamiltonian circuits in $G$, then TDTP requires the finding of $x^* \in H$ for which $C(x^*) \le C(x)$, $(\forall) x \in H$. TSP is a special case of TDTP corresponding to $d(i) = 0$, $i = 0,..., n$ and since TSP is NP-hard, TDTP is NP-hard, too.

### 3. Branch & bound technique for TDSP-an outline

In this section, the main components of a branch & bound approach for TDSP that are used for hybridizing a genetic algorithm are presented.

#### 3. 1. Outline of the technique

The branch & bound technique ([8]) requires the organization of the space of solutions $H$ as a tree, denoted by $T$. The root $r$ of $T$ corresponds to $H$ whereas each terminal node of $T$ corresponds to an individual solution $x \in H$. A node $v$ which is neither terminal nor the root of $T$ is associated with a simple

ISBN 978-3-901608-32-2

path $x = [x_1,...,x_k]$ in $G$ and corresponds to the subset $H(x)$ of all hamiltonian circuits having $x = [x_1,...,x_k]$ as prefix of length $k$. The branch & bound technique uses an estimate function $e: T \to Z^+$ that satisfies $e(x) = C(y)$, when $\{y\} = H(x)$ and x is terminal node in $T$, and $e(x) \le \min\{C(y) / y \in H(x)\}$ for any non-terminal node. The branching rules allow the growing of $T$ starting from the root and according to the least cost strategy that indicates to branch a node $v$ with the prefix $x$ if $e(x)$ reaches the minimum of estimate function values computed for all active nodes of $T$. A descendent of $v$ corresponds to an extension $x' = [x_1,...,x_k, z]$ of the prefix $x$ with exactly one vertex $z$ which does not belong to $x$ and is accessible from $x_k$. Then, node $v$ is deleted from the set of active nodes and its children become active nodes.

The search stops when a terminal node in $T$ has the minimum estimate function values among the active nodes in $T$.

**3.2. Defining the estimate function**

Consider a prefix $y = [y_1,..., y_k]$ of the set $H(y)$ of hamiltonian circuits. In order to have a more accurate estimation of the minimum cost of solutions in $H(y)$, the estimation function $e$ is taken as

$$e(y) = e_1(y) + e_2(y),$$

where $e_1(y)$ is the transportation cost on the path $[0, y_1,..., y_k]$ and $e_2(y)$ is a lower bound of the cost for supplying the consumers $V \setminus \{y_1,..., y_k\}$ and the returning at "0". Clearly,

$$e_1(x) = \cos t(D; 0, y_1) +$$

$$\sum_{i=1}^{k-1} \text{cost}(D - \sum_{h=1}^{i} d(h); y_i, y_{i+1}).$$

The way to obtain lower bounds used to define the term $e_2$ of the estimate function is described below.

*3.2.1. Transporter oriented lower bound*

Let $y = [y_1,..., y_k]$ the prefix under evaluation and consider a permutation of the consumers having $y$ as prefix, $\overline{y} = [y_1,..., y_k, z_{k+1},..., z_n]$. Denote by $q(y) = d(0) + d(z_{k+1}) + ... + d(z_k)$ the charge to be transported from $y_k$ for delivering (in any order) the demands of $z_{k+1},..., z_n$.

The following cases are considered:

(i). For $k = n$ define $e_2'(y) = d(0) \cos t(z_n, 0)$.

(ii). If $k = n-1$ compute

$$e_2'(y) = [d(0) + d(z_n)]\text{cost}(y_{n-1}, z_n) + \\ + d(0)\text{cost}(z_n, 0).$$

(iii). If $k = n-2$ take

$$\begin{aligned} e_2'(y) = \min\{ & (d(0) + d(z_{n-1}) + \\ & + d(z_n))\text{cost}(y_k, z_{n-1}) + \\ & + \left(d(0) + d(z_n)\right)\text{cost}(z_{n-1}, z_n) + \\ & + d(0)\text{cost}(z_n, 0), \\ & \left(d(0) + d(z_{n-1}) + d(z_n)\right)\text{cost}(y_k, z_n) + \\ & + \left(d(0) + d(z_{n-1})\right)\text{cost}(z_n, z_{n-1}) + \\ & + d(0)\text{cost}(z_{n-1}, 0)\} \end{aligned}$$

(iv). For $k < n-2$ the following procedure is applied. Compute a lower bound of the costs to enter the set of consumers $V(y) = V \setminus \{y_1,..., y_k\} \cup \{0\}$ given by

$$e_2^{(1)}(y) = q(y) \min\{\text{cost}(y_k, z_j) / j = k+1,..., n\}$$

and a lower bound of the costs to leave $V(y)$ for returning at "0"

$$e_2^{(3)}(y) = d(0) \min\{\text{cost}(z_j, 0) / j = k+1,..., n\}.$$

In order to obtain a lower bound of the costs for moving inside $V(y)$, consider that $\xi_{k+1},..., \xi_{k+(n-k)}$ is a permutation of $V(y)$ that sorts the consumers in the descending order of their demands, namely $d(\xi_{k+1}) \ge d(\xi_{k+2}) \ge ... \ge d(\xi_{k+(n-k)})$. Denote by $w_1,..., w_{n-k-1}$ the smallest $n-k-1$ unitary costs among the intermediate costs, $\text{cost}(z_j, z_h)$, $j, h = k+1,..., n$, $j \ne h$. Then a lower bound of the transportation costs inside $V(y)$ is given by

$$e_2^{(2)}(y) = \sum_{j=1}^{n-k-1} (q(y) - \sum_{h=1}^{j} d(\xi_{k+h}))w_j.$$

The way to construct of the lower bound $e_2^{(2)}(y)$ is shown in Fig. 2.

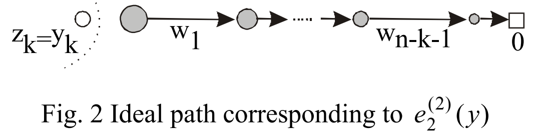


Fig. 2 Ideal path corresponding to $e_2^{(2)}(y)$

Finally, for $k < n-2$, it is obtained

$$e_2'(y) = e_2^{(1)}(y) + e_2^{(2)}(y) + e_2^{(3)}(y).$$

The parts of the total estimated cost addressed to different lower bounds are depicted in Fig.3.

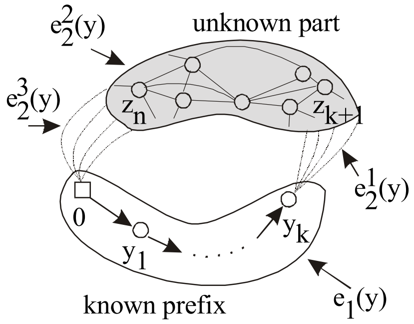


Fig. 3 Lower bounds for different segments of the delivering path

*3.2.2. Consumer oriented lower bound*

For $n-2 \le k \le n$, this bound is computed bu applying the procedure described in section 3.2.1. For $k < n-2$, let us consider again the permutation $\bar{y} = [y_1,..., y_k, z_{k+1},..., z_n]$ having the prefix $y = [y_1,..., y_k]$. The cost of the path $[z_k = y_k, z_{k+1},...z_n, 0]$ is

$$c_2(y) = \sum_{j=k+1}^{n} \text{cost}(z_{j-1}, z_j)(q(x) - \sum_{h=k+1}^{j-1} d(z_{j-1})) + \\ +\text{cost}(z_n, 0)d(0)\ .$$

Rearranging the terms in $e_2(x)$ and taking $z_{n+1} = 0$, it is obtained that

$$c_2(y) = \sum\nolimits_{j=k+1}^{n+1} d(z_j) \sum\nolimits_{h=k}^{j-1} \text{cost}(z_h, z_{h+1})\ .$$

Denote by $c^*(i, j)$ the cost of shortest path from $i$ to $j$ in $G$, where the weights of edges are the unitary costs. Since $[z_k = y_k, z_{k+1},...z_j]$ is a path from $y_k$ to $z_j$, it follows that

$$\sum\nolimits_{h=k}^{j-1} \cos t(z_h, z_{h+1}) \ge c^*(y_k, z_j),\ j = k+1,..., n+1\ .$$

Therefore,

$$e_2'' = \sum\nolimits_{j=k+1}^{n+1} d(z_j) c^*(y_k, z_j)$$

is a consumer-oriented lower bound of the solutions having $y$ as prefix. The meaning of $e_2''$ is explained in Fig.4.

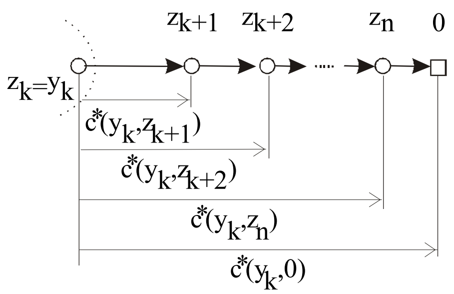


Fig. 4 The structure of the lower bound $e_2''$

*3.2.3. A simplier lower bound*

Further, a new lower bound to $c_2(x)$ is described for $k < n-2$. This bound is less accurate but it requires a smaller amount of computing time.

Compute

$\alpha = e_2^1(y)$,

$\beta = \min\{\text{cost}(z_j, z_h)/ j, h = k+1,..., n, j \ne h\}$

and

$\gamma = e_2^3(y)$.

If $\xi_{k+1},..., \xi_n$ is a permutation of $V(y)$ that sorts the remaining consumers in descending order of their demands, then it is obvious that

$$c_2(y) \ge \sum\nolimits_{j=k+1}^{n} d(\xi_j)\left[\alpha + \beta(j-k-1)\right] + \\ + d(0)\left[\alpha + \beta(n-k-1) + \gamma\right] = e_2'''(y)\ .$$

Finally,

$$e_2(y) = \max\{e_2'(y), e_2''(y), e_2'''(y)\}$$

defines a lower bound which is closer to the minimum cost than each individual bound $e'$, $e''$, $e'''$.

## 4. Memetic algorithm

The main components of the combination of a genetic algorithm ([9]) with the above outlined branch & bound method are described below.

### 4.1. Solution representation

Since branch & bound methods operate with partial solutions corresponding to the nodes of the state space tree, the chromosome is a simple path $[0, x_1,..., x_k]$ in $G$ where $x_1,..., x_k \in V \setminus \{0\}$, $1 \le k \le n$, which is simply denoted by $x = [x_1,..., x_k]$.

As mentioned before, $H(x) \subset H$ is the set of all hamiltonian circuits having the prefix $x = [0, x_1,..., x_k]$. The memetic algorithm operates with sets $H(x)$ of incompletely specified solutions instead of individual solutions. The chromosome $x$ with $k < n$ is called embryonic whereas for $k = n$ it is an adult chromosome.

### 4.2. Initial population

Four sources are used to create the initial population. First, $p_1$ percent of initial population is randomly by generated.

If geographical distance is available, then another percent $p_2$ of individuals are generated using a mobile vicinity $N(v, R_0)$ of center $v$ and radius $R_0$. One starts from “0” and $y_1$ is randomly selected from $N(0, R)$. Then the center of the vicinity is moved to

$y_1$ and $y_2$ is selected from $N(y_0, R_0)$, and so on, until the length of the prefix is reached.

Under the same assumption, a percent $p_3$ of individuals are generated to have an imprint of the clustering information contained by the set of consumers. For this, a random number $K$ in the range $[4, n/5]$ is chosen and a clustering of $V$ with $K$ clusters $C_1,...,C_K$ as produced using a simple algorithm like $c$-means [10]. Denote by $w_h$ the center of $C_h$, $h = 1,...,K$. The clusters are relabeled such that $0 \in C_1$, and

$$((\|w_h - w_{h+1}\| \le \|w_h - w_j\|, j = h+1,...,n),$$
$$h = 1,...,n-1),$$

where $\|.\|$ denotes the Euclidean distance in $R^2$.

Then, for a given length of a chromosome, one consider random permutations of clusters $C_1, C_2, ...$ and these permutations are concatenated (in this order) to form the initial individuals capturing the clustering information.

A percent $p_4$ of the initial individuals is obtained from the optimal paths corresponding to the optimal costs $c^*(0, j)$, $j = 1,...,n$. Since these optimal paths are simple, they can be used as a genetic material. For the sake of efficiency, only the optimal path containing more then $n/10$ arcs are included in the initial population. The Floyd-Warshall algorithm [11, p.558] was applied to find the optimal costs and the corresponding optimal paths.

Appropriate values of these percents are: $p_1 \cong 10\%$, $p_3 \cong 10-15\%$, if $n \ge 50$ then $p_4 \cong 5-10\%$ otherwise $p_4 \cong 10-20\%$, and finally, $p_2 = 100 - (p_1 + p_3 + p_4)$.

### 4.3. Fitness function

A branch & bound algorithm explores a population of candidate partial solutions that covers the entire set of solution and guides its search using the estimate function. Therefore, it is natural and consistent with the optimization goal to use the estimate function as a fitness function. On the other hand, observe that during the first stages of the search when the length of embryos is small, $e_2$-part of estimate function plays a major role, whereas toward the end of the search $e_1$-part is decisive in the guiding the search because the contribution of $e_2$ is rather modest. In order to reinforce this behavior it is used fitness function, defined by

$$fit(x) = \frac{k}{n} e_1(x) + \frac{n-k}{n} e_2(x),$$

where $x = [x_1,...,x_k]$, $1 \le k \le n$. Remark that $\min\{e_1(x), e_2(x)\} \le fit(x) \le \max\{e_1(x), e_2(x)\}$ and for $k = n$ $fit(x) = e_1(x)$, thus $fit(x)$ satisfies the requirements imposed to an estimate function.

### 4.4. Genetic operators

A combination between the usual mutual exchange operator and the vicinity rule is adopted as mutation operator. A random vertex $v$ in the chromosome is selected. Afterwards, a second vertex $u$ is randomly chosen from the remaining positions of the chromosome that belong to $N(v, R)$. Finally, $u$ and $v$ are interchanged. Let $r_0$ be the minimum distance between two consumers. The value of $R$ decreases from $R_0$ to $r_0$ as the evolution advances. The action of the mutation operator is illustrated in Fig. 5.

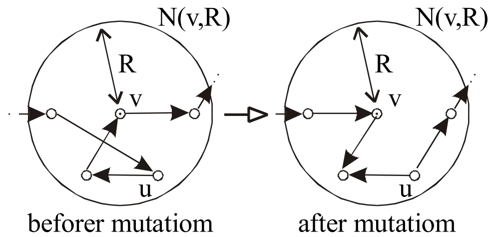


Fig. 5 Interchanging of nodes $v$ and $u$ by mutation

Due to the specific of problem, the crossover must preserve the relative order of genes. It was adopted the mixed strategy to use one point crossover for the first half of the evolution stages and a two-point crossover for the remaining stages. Whenever such a mutation is applied, excepting the first segment of genes, every remaining segment is completed so that each offspring is a simple path.

Since in the final of the search process a fully specified solution must be found, a growing operator is successively applied for completing an embryo until it become adult. If $x = [x_1,...,x_k]$ is the path to be completed with $x_{k+1}$ then

$$x_{k+1} \in N(x_k, R) \setminus \{x_1,...,x_k\}$$

and

$$\text{cost}(x_k, x_{k+1})(q(x_k) - d(x_{k+1})) \le$$
$$\text{cost}(x_k, v)(q(x_k) - d(v)),\ (\forall) \in N(x_k, R) \setminus \{x_1,...,x_k\}.$$

This choice is suggested by a greedy approach. This approach uses as optimizations measure the minimization of the cost of the charge that is not used for supplying the added node. If no selection is possible then $R$ must be higher. This choice is illustrated in Fig.6.

 

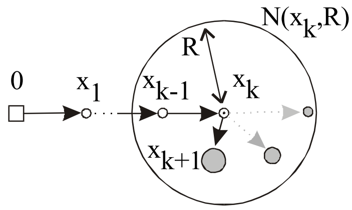


Fig. 6 Adding $x_{k+1}$ to the prefix $[x_1,...,x_k]$

Mutation, crossover and growing are applied with prescribed probabilities $\pi_m$, $\pi_c$ and $\pi_g$ respectively.

### 4.5. Population management

The mating pool consists in the first 40% of the best individuals. Since the population uses limited memory resources, the numbers of chromosomes can be higher when their lengths are smaller. The hybrid algorithm uses a decreasing population size and is based on a generational approach. The new individuals generated by mutation, crossover and growing are evaluated by the fitness function. An elitist selection is applied to decide the individuals entering the next generation. The search stops when the population has only adult chromosomes and the average of fitness values stagnates for a prescribed number of successive generations.

### 5. Parameters tuning and performance evaluation

The tests were made using a generator of random instances having an apparent clustering structure.

The experiments showed that the appropriate values of probabilities of genetic operators are $\pi_m \approx 0.1$, $\pi_c = 0.25 - 0.3$ and $\pi_g = 0.1 - 0.15$. A decrease of the radius $R$ from half of the maximal distance between two consumers and $r_0$ along 1000 of evolution stages produced better results than a constant value of $R$.

It was experimentally shown that capturing the intrinsic clustering information of data improves the results with about 5-8% and accelerates the search.

Another group of tests executed on artificial instances with apparent optimal solution showed that, the better the clusters separation, the closer the results will be to the optimal solutions. The costs of the found solutions are 5-8% greater than the optimal costs.

Another interesting observation is that the algorithm could be stopped as soon as an adult chromosome having the best fitness value was found since the improvement produced after this moment are in general modest. For problem instances having between 50 and 110 consumers, the algorithm consumed 500-700 generation before stopping its activity. The starting value of the population size was 300 of individuals whereas the final value was 50.

The solutions produced by this algorithm are with 10-15% better than those produced by the techniques in [3].

### 6. Conclusions and further research

The paper presented a new hybrid genetic algorithm resulted from a combination with a branch & bound technique for solving a transportation problem with variable charge along hamiltonian circuits. The proposed algorithm has a good performance with respect to the quality of solutions and the computing time.

Further research concerns the design of a segregative version of the hybrid algorithm that is able to exploit better the existence of multiple local optima.

## References

[1] M. Boulif and K. Atif , A new branch-&-bound-enhanced genetic algorithm for the manufacturing cell formation problem, *Computers and Operations*, vol. 33 , no. 8 , August 2006, pp. 2219 - 2245

[2] O. Brudaru and B. Valmar, Genetic Algorithms with Embryonic Chromosomes for Assembly Line Balancing with Fuzzy Processing Times, *8th International Research/Expert Conference: Trends in the Development of Machinery and Associated Technology*, TMT 2004, Neum, Bosnia and Herzegovina, 15-19 Sept. 2004, pp.891-894.

[3] O. Brudaru and A. Vîlcu, Genetic Algorithm with Accelerating Hybrid Components for Affine Cost Hamiltonian Circuits, *ICPR-16, 16th International Conference on Production Research*, 30 July-3 August, 2001, Prague.

[4] O.Brudaru and A. Vîlcu, A., Genetic algorithm for a transportation problem with variable charge along hamiltonian circuits, 10th *International DAAAM Symposium - Intelligent Manufacturing & Automation: Past - Present - Future*, Oct. 21-23 1999, Vienna, Austria.

[5] W. Domschke, *Logistik: Rundreisen und Touren* (Bd.2), 2, Aufl., Oldenbourg, Munchen, 1989

[6] A. P. French, A. C. Robinson and J. M. Wilson, Using a Hybrid Genetic-Algorithm/Branch and Bound Approach to Solve Feasibility and Optimization Integer Programming Problems, *Journal of Heuristics*, vol. 7, no. 6 , Nov. 2001, pp. 551-564

[7] M. Hiroshi and S. Naoki, Hybrid Branch and Bound Method with Genetic Algorithm for

Flexible Flowshop Scheduling Problem, *JSME International Journal Series C*, Vol. 48, No. 1, 2005, pp.46-52.

[8] E. Horowitz and S. Sahni, *Algorithm: design and analysis*, Computer Science Press, Inc., 1982

[9] Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolution Program*, 2nd ed., Springer Verlag, Berlin, 1994.

[10] R. Rojas, *Neural Networks - A Systematic Introduction*, Springer Verlag, Berlin, 1996.

[11] T. Cormen, C. E. Leiserson and R. L. Rivest, *Introduction to Algorithms*, 1st edition, MIT Press and McGraw-Hill,, 1990.

*ISBN 978-3-901608-32-2*  