International Journal for Traffic and Transport Engineering, 2014, 4(3): 286 - 296

DOI: http://dx.doi.org/10.7708/ijtte.2014.4(3).04 UDC: 656.022.5

LOCATING HUBS IN TRANSPORT NETWORKS: AN ARTIFICIAL INTELLIGENCE APPROACH

Dušan Teodorović[1], Milica Šelmić[2], Ivana Vukićević[3]

[1, 2, 3] *University of Belgrade, Faculty of Transport and Traffic Engineering, Vojvode Stepe 305, 11000 Belgrade, Serbia*

Received 6 May 2014; accepted 5 August 2014

Abstract: Hub facilities serve as switching and transshipment points in transportation and communication networks as well as in logistic systems. Hub networks have an influence on flows on the hub-to-hub links and ensure benefit from economies of scale in inter-hub transportation. The key factors for designing a successful hub-and-spoke network are to determine the optimal number of hubs, to properly locate hubs, and to allocate the non-hubs to the hubs. This paper presents the model to determine the locations of the p-hub facilities in the network and to allocate the non-hubs to the hubs. The problem is solved by the Bee Colony Optimization (BCO) algorithm, and the results are compared with the optimal solutions obtained by CPLEX. The BCO algorithm belongs to the class of stochastic swarm optimization methods. The proposed algorithm is inspired by the foraging habits of bees in the nature. The BCO algorithm was able to obtain the optimal value of objective functions in all test problems. The CPU times required to find the best solutions by the BCO are acceptable.

Keywords: hub problem, location theory, Bee Colony Optimization.

1. Introduction

Modern transportation networks are not fully connected. A great number of passengers, containers and parcels are transported from one node to another without a direct service. Air carriers, freight operators, and delivery companies usually organize a hub transportation network as the flows between hubs are characterized by economies of scale. At hubs, passengers are changing aircraft, while containers and parcels are exchanged across, trucks, vessels and planes. When planning and organizing their activities, transportation operators should try to find the answers to the following questions: (i) What should be the total number of hubs? (ii) Where should these hubs be located? (iii) How should demand for the hubs' services be allocated to the hubs? The origin-destination (O-D) matrix that contains information about flows between all pairs of nodes in the network represents the key input data for the hub location problem. Hub networks could be characterized by the single or multiple allocations. In the case of single allocation, all the incoming and outgoing traffic in demand center is routed through a single hub. In the case of multiple allocation, demand nodes can incoming and outgoing traffic routed through more than one hub. There exist different variants of hub location problems such as p-hub location problem, p-hub median location problem, p-hub location problem with limited capacity, p-hub center location problem, p-hub maximal covering location problem, etc.

[2] Corresponding author: m.selmic@sf.bg.ac.rs

In this paper, we study a single allocation hub-and-spoke network design problem. We analyze the case in which there are no capacity constraints at hubs. *P*-hub problems belong to the class of *NP*-hard combinatorial optimization problems. Computational complexity of the *p*-hub problems forced many researchers to propose and use various heuristic and meta-heuristic algorithms. Meta-heuristics have become main tool for solving hard combinatorial optimization problems. In the greater part of cases, meta-heuristics offer high-quality solutions within rational CPU time. Among meta-heuristics, a group of biologically inspired algorithms is known. Bee colony optimization (BCO) method, that uses collective intelligence applied by the honey bees during nectar collecting process, is one of them. BCO has been proposed by Lučić and Teodorović (2001, 2003) and up to now it is successfully applied to a range of real-life optimization problems.

This fact motivated the authors to use the Bee Colony Optimization (BCO) meta-heuristic technique as appropriate tool. BCO is stochastic, effective meta-heuristic method and it has been successfully applied to many combinatorial optimization problems, mostly in transport, location and scheduling fields. The proposed model is supported by numerical examples on real data related to the parcel delivery network in Turkey. The obtained results show that the proposed meta-heuristic can generate high-quality results within acceptable CPU times.

The paper is organized in the following way. The statement of the problem is given in the Section 2. The Section 3 is devoted to the BCO technique. The BCO approach to the *p*-hub location problem is given in the Section 4. The test problems are given in Section 5. Conclusions are given in the Section 6.

2. Mathematical Formulation

The hub concept refers to the strategically located facilities (points) in which logistic companies and carriers organize reloading of goods, since flows between hubs are characterized by economies of scale effect. At hubs, goods are exchanged across vans, trucks, and planes. The system organized through hubs allows the operation of larger transport equipment with higher frequency between hubs. The consequence of such an organization is lower costs per unit of goods. On the other hand, the time of the goods' transport is higher (Teodorović, 2007).

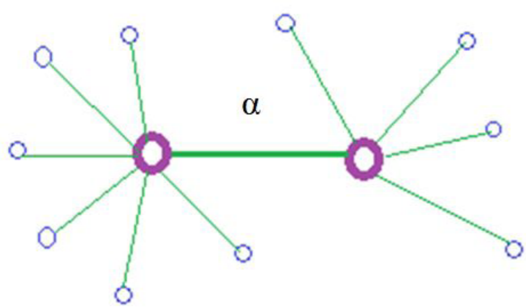

Fig. 1.
An Example of Hub Network with Single Allocation

O'Kelly (1987) developed the first mathematical formulation for a hub location problem by studying airline passenger networks. The author also presented a data set based on the airline passenger interactions between 25 US cities in 1970 (Civil Aeronautics Board (CAB) data). O'Kelly's formulation is related to the single allocation p-hub median problem that minimizes the total transportation cost of the demand flows. Each non-hub node must be allocated to just one hub node. The number of hubs to locate is defined exactly and denoted by p, and at least one or at most two hub nodes have to be traversed when traveling between two non-hub nodes. Furthermore, there is no cost for establishing hubs and the hubs are uncapacitated.

To reflect the economies of scale in hub-to-hub connections, O'Kelly (1987) introduced a constant discount factor, $0 \leq \alpha \leq 1$, for using inter-hub connections. O'Kelly (1987) studied the hub location problem in a case of non-oriented network, represented by a graph $G = (N, A)$. This graph includes the set of consecutively numbered nodes N, as well as the set of the consecutively numbered links A. W_{ij} denotes the number of flow units between node i and node j, while C_{ij} is transport cost per unit between node i and node j.

Let introduce binary variable x_{ij}:

$$x_{ij} = \begin{cases} 1, & hub\ \ located\ \ in\ \ node\ \ j\ \ servers\ \ clients\ \ from\ \ node\ \ i \\ 0, & otherwise \end{cases}$$

The following is mathematical formulation suggested for the single allocation p-hub median problem (O'Kelly, 1987):

$$\min \sum_i \sum_j W_{ij} (\sum_k X_{ik} C_{ik} + \sum_m X_{jm} C_{jm} + \alpha \sum_k \sum_m X_{ik} X_{jm} C_{km} \quad (1)$$

Subject to:

$$(n-p+1) X_{kk} - \sum_i X_{ik} \geq 0 \qquad \forall k \qquad (2)$$

$$\sum_k X_{ik} = 1 \qquad \forall i \qquad (3)$$

$$\sum_k X_{kk} = p \qquad (4)$$

$$X_{ik} \in \{0,1\} \qquad \forall i, \qquad \forall k \qquad (5)$$

The proposed objective function minimizes the total transportation cost of the demand flows. Constraint (2) enables that node has to be allocated to k only if k is a hub node. Constraint (3) guarantees that each node is allocated to a hub. The number of hubs to be located is fixed to p by constraint (4). The last constraint refers to the binary nature of the variables.

3. Related Work

The key factors for designing a successful hub-and-spoke network are to determine the optimal number of hubs, to properly locate hubs, and to allocate the non-hubs to the hubs.

The first heuristic presented in O'Kelly (1987) allocates each node to the nearest hub.

The second heuristic allocates each node to its first or second nearest hub. Heuristic described in Klincewicz (1992) are based on tabu search (TS) and a greedy randomized adaptive search procedure (GRASP).

Chen (2007) studied the uncapacitated single allocation hub location problem. He proposed two approaches to determine the upper bound for the number of hubs, as well as hybrid heuristic (based on the simulated annealing method, tabu list, and improvement procedures) to resolve the uncapacitated single allocation hub location problem.

Kratica et al. (2007) proposed two genetic algorithm (GA) approaches. The numerical experiments were carried out on the standard ORLIB hub data set. Two evolutionary algorithms (EAs) that use binary encoding and standard genetic operators adapted to the problem were used in Kratica et al. (2011).

Four variations of the ant colony optimization meta-heuristic are developed in Randall (2008). The results reveal that each of the approaches can return optimal solution costs in a reasonable amount of computational time.

To solve hub and spoke system design, Elhedhli and Wu (2010) proposed a Lagrangean heuristic where the problem is decomposed into an easy sub-problem and a more difficult nonlinear sub-problem. Ilić et al. (2010) presented a new general variable neighborhood search approach for the uncapacitated single allocation *p*-hub median problem. Variable neighborhood search and path relinking have been also proven to be very effective in solving *p*-hub problems (Pérez et al., 2007).

Calık et al. (2009) presented an efficient heuristic based on tabu search and test the performance of the heuristic on the CAB data set, as well as on the Turkish network. Čupić and Teodorović (2012) explored parcel express service that is associated with the *p*-hub problem. In their study, the authors used Genetic algorithm.

4. Bee Colony Optimization

The Bee Colony Optimization (BCO) meta-heuristic was introduced by Lučić and Teodorović (2001, 2003) as a new direction in the field of Swarm Intelligence. The BCO algorithm is inspired by the foraging behavior of honeybees. The basic idea behind the BCO is to build a multi-agent system (a colony of artificial bees) that can efficiently solve hard combinatorial optimization problems. The artificial bee colony behaves similarly to bee colonies in nature in some ways but differently from them in other ways.

4.1. The BCO Algorithm

During the evolution of the BCO algorithm authors developed two different approaches. The first approach is based on the constructive steps in which bees build solutions step by step. The second and very actual approach of the BCO algorithm is based on the improvement of complete solutions in order to obtain the best possible final solution. In this paper we use this concept.

The BCO is a population based algorithm. Population of *artificial bees* searches for the optimal solution. Every artificial bee generates one solution to the problem. The algorithm consists of two alternating phases: *forward pass* and *backward pass*. During each forward pass, every bee is exploring the search space. It applies a predefined number of moves, which construct and/or improve the solution, yielding to a new solution.

Having obtained new solutions, the bees go back to the nest and begin the second phase, the so-called backward pass. During the backward pass, all bees share information about their solutions. In nature, bees would carry out a dancing ritual, which would report to other bees about the amount of food they have found, and the closeness of the patch to the nest. In the search algorithm, the bees make known the quality of the solution, i.e. the value of the objective function. During the backward pass, every bee decides with a certain probability whether it will promote its solution or not. The bees with better solutions have more chances to advertise their solutions. The remaining bees have to decide whether to continue to explore their own solution in the next forward pass, or to begin exploring the neighborhood of one of the solutions being advertised. In the same way, this decision is taken with a probability, so that better solutions have higher probability of being chosen for exploration.

The two phases of the search algorithm, forward and backward pass, are performed *iteratively*, until a stopping condition is met. The possible stopping conditions could be, for example, the maximum total number of forward/backward passes, the maximum total number of forward/backward passes without the improvement of the objective function, etc.

The BCO algorithm parameters whose values need to be set prior the algorithm execution are as follows:

B – the number of bees involved in the search and

NC – the number of constructive/ improvement moves.

The pseudo-code of the BCO algorithm could be described in the following way:

Do

1. Initialization: a(n) (empty) solution is assigned to each bee.

2. For ($i = 0$; $i < NC$; i ++)

//**forward pass**

(a) For ($b = 0$; $b < B$; b++)

i) Evaluate possible moves.

ii) Choose one move using the roulette wheel.

//**backward pass**

(b) For ($b = 0$; $b < B$; b++)

Evaluate the partial/complete solution for bee b;

(c) For ($b = 0$; $b < B$; b++)

Loyalty decision using the roulette wheel for bee b;

(d) For ($b = 0$; $b < B$; b++)

If (b is uncommitted), choose a recruiter by the roulette wheel.

3. Evaluate all solutions and find the best one.

while stopping criteria is not satisfied.

Steps 1, (a) and (b) are problem dependent and should be resolved in each BCO implementation. On the other hand, there are formulae specifying steps (c), loyalty decision, and (d), recruiting process, and they are described in the next section.

5. The BCO Approach to P-Hub Problem

In this section, we apply the BCO algorithm to the p-hub problem. The BCO improvement version is more suitable due to the quadratic nature of the objective function. In contrast to the constructive version, in the BCO improvement version complete solutions are assigned to the bees at the beginning, and modify through the iterations. Up to now this concept of BCO was successfully used in the relevant literature for solving the *p*-center problem (Davidović et al., 2011) and the transit network design problem (Nikolić and Teodorović, 2012).

The objective function to be minimized represents the total transportation costs of the demand flows. The flow between each two non-hub nodes in the network, *i* and *j*, passes through two hubs.

The first step of the algorithm, called preprocessing, is performed off-line, while all other steps are online steps. In this step the input data is transformed in order to reduce the time required for all computations performed online. Each element of the W_{ij} matrix is multiplied with appropriate element of the matrix C_{ij}. In such a way the matrix W_{ij} x C_{ij} is formed. Therefore, this matrix contains information about total transportation costs between the nodes. Finally, we summarize the elements in each matrix row. The sum $\sum C_{ij} W_{ij}$ represents the total transportation costs of flows traveling from all nodes to hub located in the node *j*.

The second step represents generation of the initial complete solution. In the third step results comparison mechanism and recruitment are performed. The bees modify current solutions in the fourth, the most significant, step. The last three steps are executed in the real CPU time.

Initial, complete solutions are formed in following way. Let us denote by *Vi* bee's utility in the case when bee chooses node *i* to be the hub. We introduce the assumption that the bee's utility depends on cost per flow unit for each node.

Based on utility V_i bees chose nodes to be hub locations. We assume in this paper that the bee's utility equals:

$$V_i = \frac{\sum_j W_{ij}}{\sum_j C_{ij} W_{ij}} \tag{6}$$

We define the probability p_i that the specific bee chooses node *i* in the following way:

$$p_i = \frac{V_i}{\sum_{k=1}^{K} V_k}, \qquad i = 1,2,\ldots,n \tag{7}$$

where:

K – the number of "free" nodes (not previously chosen).

Obviously, nodes with a lower cost per flow unit have a higher chance to be chosen. Using Eq. (7) and a random number generator, we determine the nodes to be chosen by each bee.

After determining hubs in current complete solution, it is necessary to allocate all other nodes to those hubs. Each node is assign to the hub for which the value C_{ij} x W_{ij} (from matrix W_{ij} x C_{ij}) is the lowest possible.

Let us denote by C_i ($i = 1,2,\ldots,b$) the total cost per flow unit in the solution generated by the *i*-th bee. Let us normalize the cost C_i. We denote by O_i normalized value of the cost C_i, i.e.:

$$O_i = \frac{C_{\max} - C_i}{C_{\max} - C_{\min}}, \quad O_i \in [0,1] \quad i = \overline{1,b} \qquad (8)$$

where $C_{\min}$ and $C_{\max}$ are respectively minimum and the maximum cost produced by all bees.

After the completion of a forward pass, each bee decides whether it stays loyal to the previously discovered solution or not. This decision depends on the quality of its own solution related to all other existing solutions. The probability that b-th bee (at the beginning of the new forward pass) is loyal to its previously generated complete solution is expressed as follows:

$$p_b^{u+1} = e^{-\frac{O_{\max} - O_b}{u}}, \quad b = 1,2,\ldots,B \qquad (9)$$

where:

O_b – denotes the normalized value for the objective function of complete solution created by the b-th bee;

$O_{\max}$ – represents the maximum over all normalized values of complete solutions to be compared;

u – counter of the forward passes (taking values 1, 2, .., NC).

For each uncommitted bee it is decided which recruiter it will follow, taking into account the quality of all advertised solutions. The probability that b's complete solution would be chosen by any uncommitted bee equals:

$$p_b = \frac{O_b}{\sum_{k=1}^{R} O_k}, \quad b = 1,2,\ldots,R \qquad (10)$$

where O_k represents the normalized value for the objective function of the k-th advertised solution and R denotes the number of recruiters. Using Eq. (10) and a random number generator, each uncommitted follower joins one recruiter through a roulette wheel.

The main step of the BCO improvement algorithm is modification of solution through NC forward passes within the single iteration. It has been proved to be the key factor that enables searching the best possible solution. We designed this step in such a way to assure different treatment of same solutions. Our modification consists of substituting some of the p hubs with nodes selected from the remaining n-p non-hub nodes.

We divide solution modification into two steps.

1) The first step consists of determining the number of hubs to be replaced, Q, $Q=1,2,\ldots,p$. This number is determined in a random manner.

2) In the second step, we remove Q locations from the hub list, and add new Q hubs. The removal and selection of particular Q hubs is done according to presented formulae (6), (7) and roulette wheel.

6. Experimental Evaluation

The proposed algorithm is tested on a various test problems related to the Turkish parcel network. The analyzed network originally consists of 81 nodes (towns). In this paper we use only the first 11 nodes. The reason for this constraint is in the fact that CPLEX can solve this problem to optimality up to 11 nodes. In Table 1 are shown the transport costs per unit and in Table 2 number of flow units.

Table 1
Transport Cost per Flow Unit

	Adana	Adiyaman	Afyon	Ağri	Amasya	Ankara	Antalya	Artvin	Aydin	Balikesir	Bilecik
Adana	0,0000	0,3890	0,5187	1,3435	1,2322	0,0899	0,2386	4,0076	0,6907	0,6115	2,9062
Adiyaman	0,3962	0,0000	2,4683	2,7216	3,8671	0,4194	1,1467	8,8628	2,8566	2,5274	12,1020
Afyon	0,5268	2,4614	0,0000	4,2257	2,7608	0,1090	0,2897	11,1272	0,6295	0,5087	1,8463
Ağri	1,3704	2,7254	4,2436	0,0000	5,2728	0,6920	2,1816	5,3449	4,5588	3,8197	18,3486
Amasya	1,2599	3,8820	2,7792	5,2856	0,0000	0,3192	1,9373	14,0452	3,7605	2,9467	12,2460
Ankara	0,0870	0,3983	0,1038	0,6562	0,3020	0,0000	0,1038	1,7084	0,2082	0,1616	0,5287
Antalya	0,2391	1,1280	0,2858	2,1430	1,8984	0,1076	0,0000	6,0605	0,2867	0,3755	1,9328
Artvin	4,1083	8,9198	11,2302	5,3717	14,0813	1,8106	6,2007	0,0000	12,2397	9,9814	47,5107
Aydin	0,7001	2,8427	0,6282	4,5303	3,7278	0,2182	0,2900	12,1023	0,0000	0,3947	3,9173
Balikesir	0,6186	2,5104	0,5067	3,7886	2,9157	0,1691	0,3791	9,8509	0,3940	0,0000	1,6183
Bilecik	2,9792	12,1794	1,8633	18,4398	12,2771	0,5603	1,9774	47,5090	3,9616	1,6397	0,0000

Source: http://ie.bilkent.edu.tr/~bkara/hub_location.php

Table 2
Number of Flow Units

	Adana	Adiyaman	Afyon	Ağri	Amasya	Ankara	Antalya	Artvin	Aydin	Balikesir	Bilecik
Adana	0	17493	22782	14827	10242	112387	48225	5382	26661	30183	5449
Adiyaman	17174	0	7544	4910	3391	37216	15969	1782	8828	9995	1804
Afyon	22429	7565	0	6412	4429	48604	20856	2328	11530	13053	2357
Ağri	14536	4903	6385	0	2871	31499	13516	1508	7472	8459	1527
Amasya	10016	3378	4400	2864	0	21706	9314	1039	5149	5829	1052
Ankara	116190	39190	51038	33217	22945	0	108040	12058	59729	67619	12208
Antalya	48130	16234	21142	13760	9505	104299	0	4995	24742	28010	5057
Artvin	5250	1771	2306	1501	1037	11377	4882	0	2699	3055	552
Aydin	26302	8872	11554	7520	5194	56998	24458	2730	0	15307	2764
Balikesir	29833	10062	13105	8529	5891	64649	27740	3096	15336	0	3135
Bilecik	5316	1793	2335	1520	1050	11520	4943	552	2733	3094	0

Source: http://ie.bilkent.edu.tr/~bkara/hub_location.php

We applied the BCO algorithm, as well as the CPLEX, and compared the obtained results. The results of comparison between our BCO and the optimal results obtained by CPLEX are summarized in Table 3. The first column of the Table 3 contains the total number of nodes. The number of hubs to be located is given in the first row.

Since in tested benchmark problems number of hubs, p, is not defined, we vary this number from 2 to 4. Also, we change parameter α from 0 to 1 with increment 0.2.

Table 3

Comparison between BCO and Optimal Results Obtained by CPLEX

			Number of hubs							
			p=2				**p=4**			
			α=0,2	**α=0,4**	**α=0,6**	**α=0,8**	**α=0,2**	**α=0,4**	**α=0,6**	**α=0,8**
Number of nodes	**10**	**Cplex**	540411,6	565301,4	590191,3	615081,1	382648,3	459175,7	535703,1	606824,3
		CPU (s)	8,128	68,765	158,075	268,93	35,443	144,091	270,084	325,8
		BCO	540411,6	565301,4	590191,3	615081,1	382648,3	459175,7	535703,1	606824,3
		CPU (s)	0.458	0.4251	0.378	0.402	0.613	0.684	0.591	0.611
	11	**Cplex**	588109,1	613409,8	638710,4	664011	427615,9	505236,6	582857,3	655513,2
		CPU (s)	123,116	273,9261	391,3678	851,2563	457,7	632,125	1152,052	1792,563
		BCO	588109,1	613409,8	638710,4	664011	427615,9	505236,6	582857,3	655513,2
		CPU (s)	0.874	0.893	0.822	0.910	0.995	0.968	0.985	1.002

The proposed BCO algorithm produced results of a very high quality. The BCO algorithm was able to obtain the optimal objective function values in all considered cases. The CPU times required to find the best solutions by the BCO are very low. In other words, the BCO was able to produce "very good" solutions in a "reasonable" computation time. The results for CPU times, shown in Table 1, are obtained for the case of $I = 100$ algorithm iterations, and $B = 5$. All the tests were performed on Intel Pentium Dual CPU T2370; 1,73 GHz; 2038 RAM.

7. Conclusion

The BCO is meta-heuristic technique created by the analogy with foraging behavior of honeybees, realizing the concepts of collective intelligence. A population of artificial bees searches for the optimal solution with cooperation that enables more efficiency and allows bees to reach the goals they could not achieve individually.

In this paper the BCO heuristic algorithm is used to tackle the p-hub problem. Authors applied the improvement concept of BCO. The proposed BCO algorithm is tested on various benchmark examples. Based on the obtained results, we can conclude that BCO is able to produce high-quality solutions within negligible CPU times. We compared the BCO performances with results obtained by the CPLEX and proved that the BCO is very competitive.

The achieved results indicate that the development of new models based on swarm intelligence principles could considerably contribute to the solution of difficult location analysis and logistic problems.

Acknowledgments

This research is supported by the Ministry of Education, Science and Technological Development of the Republic of Serbia, Grant No. 36002.

References

Calık, H.; Alumur, S.; Kara, B.; Karasan, O. 2009. A tabu-search based heuristic for the hub covering problem over incomplete hub networks, *Computers & Operations Research*. DOI: http://dx.doi.org/10.1016/j.cor.2008.11.023, 36(12): 3088-3096.

Chen, J.F. 2007. A hybrid heuristic for the uncapacitated single allocation hub location problem, *Omega*. DOI: http://dx.doi.org/10.1016/j.omega.2005.05.004, 35(2): 211-220.

Čupić, A.; Teodorović, D. 2012. A Multi - objective approach to the parcel express service delivery problem, *Journal of advanced transportation*. DOI: http://dx.doi.org/10.1002/atr.1218, 2042-3195.

Davidović, T.; Ramljak, D.; Šelmić, M.; Teodorović, D. 2011. Bee colony optimization for the p-center problem, *Computers & Operations Research*. DOI: http://dx.doi.org/10.1016/j.cor.2010.12.002, 38(10): 1367-1376.

Elhedhli, S.; Wu, H. 2010. A Lagrangean Heuristic for Hub-and-Spoke System Design with Capacity Selection and Congestion, *INFORMS Journal on Computing*. DOI: http://dx.doi.org/10.1287/ijoc.1090.0335, 22(2): 282-296.

Ilić, A.; Urošević, D.; Brimbergc, J.; Mladenović, N. 2010. A general variable neighborhood search for solving the uncapacitated single allocation p-hub median problem, *European Journal of Operational Research*. DOI: http://dx.doi.org/10.1016/j.ejor.2010.02.022, 206(2): 289-300.

Klincewicz, J. 1992. Avoiding local optima in the p-hub location problem using tabu search and grasp, *Annals of Operations Research*. DOI: http://dx.doi.org/10.1007/BF02060483, 40(1): 283-302.

Kratica, J.; Milanović, M.; Stanimirović, Z.; Tošić, D. 2011. An evolutionary-based approach for solving a capacitated hub location problem, *Applied Soft Computing*. DOI: http://dx.doi.org/10.1016/j.asoc.2010.05.035, 11(2): 1858-1866.

Kratica, J.; Stanimirović, Z.; Tošić, D.; Filipović, V. 2007. Two genetic algorithms for solving the uncapacitated single allocation p-hub median problem, *European Journal of Operational Research*. DOI: http://dx.doi.org/10.1016/j.ejor.2006.06.056, 182(1): 15-28.

Lučić, P.; Teodorović, D. 2001. Bee System: Modeling Combinatorial Optimization Transportation Engineering Problems by Swarm Intelligence. *Preprints of the TRISTAN IV Triennial Symposium on Transportation Analysis*, Sao Miguel, Azores Islands, Portugal. 441-445.

Lučić, P.; Teodorović, D. 2003. Computing with Bees: Attacking Complex Transportation Engineering Problems, *International Journal on Artificial Intelligence Tools*. DOI: http://dx.doi.org/10.1142/S0218213003001289, 12(3): 375-394.

Nikolić, M.; Teodorović, D. 2013. Transit network design by Bee Colony Optimization, *Expert Systems with Applications*. DOI: http://dx.doi.org/10.1016/j.eswa.2013.05.002, 40(15): 5945-5955.

O'Kelly, M.E. 1987. A quadratic integer program for the location of interacting hub facilities, *European Journal of Operational Research*. DOI: http://dx.doi.org/10.1016/S0377-2217(87)80007-3, 32(3): 393-404.

Pérez Pérez, M.; Rodríguez, F.; Moreno-Vega, M. 2007. A hybrid VNS–path relinking for the p-hub median problem, *IMA J Management Math*. DOI: http://dx.doi.org/10.1093/imaman/dpm013, 18(2): 157-171.

Randall, M. 2008. Solution approaches for the capacitated single allocation hub location problem using ant colony optimization, *Computational Optimization and Applications*. DOI: http://dx.doi.org/10.1007/s10589-007-9069-1, 39(2): 239-261.

Teodorović, D. 2007. *Transportne mreže (Transportation networks)*. Univerzitet u Beogradu, Saobraćajni fakultet, Beograd (in Serbian) 428 p.