Transit network design by Bee Colony Optimization

CrossMark

Miloš Nikolić, Dušan Teodorović *

Faculty of Transport and Traffic Engineering, University of Belgrade, Vojvode Stepe 305, 11000 Belgrade, Serbia

A B S T R A C T

The transit network design problem is one of the most significant problems faced by transit operators and city authorities in the world. This transportation planning problem belongs to the class of difficult combinatorial optimization problem, whose optimal solution is difficult to discover. The paper develops a Swarm Intelligence (SI) based model for the transit network design problem. When designing the transit network, we try to maximize the number of satisfied passengers, to minimize the total number of transfers, and to minimize the total travel time of all served passengers. Our approach to the transit network design problem is based on the Bee Colony Optimization (BCO) metaheuristics. The BCO algorithm is a stochastic, random-search technique that belongs to the class of population-based algorithms. This technique uses a similarity among the way in which bees in nature look for food, and the way in which optimization algorithms search for an optimum of a combinatorial optimization problem. The numerical experiments are performed on known benchmark problems. We clearly show that our approach, based on the BCO algorithm, is competitive with other approaches in the literature, and it can generate high-quality solutions.

© 2013 Elsevier Ltd. All rights reserved.

1. Introduction

Urban road networks in a lot of countries are extremely congested. The consequences are high travel times, unforeseen delays, increased travel costs, increased air pollution, noise level, and number of traffic accidents. Transportation engineers and city authorities have developed and implemented various Travel Demand Management (*TDM*) techniques that increase travel choices to travelers ("Park-and-Ride facilities", "High Occupancy Vehicle (HOV) facilities", "Ride-sharing programs", "Telecommuting", "Congestion Pricing"). Still, the raising of the modal share of public transit in the cities is one of the major activities to be performed by traffic planners and city authorities. This could be done by proper design of public transit networks, expansion of existing lines and park and ride spaces, increasing the availability of direct service among origin–destination pairs, frequencies increase, development of the bus systems separated from the rest of the traffic network, transit service on nights and weekends, improving passengers' comfort and schedule reliability, better information systems for passengers (visual terminals and broadcasting information), etc.

Properly designed public transit network can significantly increase public transport mode share. The public transit network design problem is one of the most significant problems faced by bus operators and city authorities in the world. This transportation planning problem belongs to the class of difficult combinatorial

optimization problem, whose optimal solution is difficult to discover. The bus network shape, as well as bus frequencies, highly depend on both passenger demand, and on the number and type of available buses (fleet size), and/or available budget. Poorly designed bus network can cause very long passengers' waiting times, and/or inexactness in bus arriving times. In addition, inadequately designed network can show high inappropriateness among the designed bus routes and paths of the majority of users.

Many of the factors that should be taken into account when designing bus network are mutually in conflict. For example, the shorter passengers waiting times, the higher the number of buses needed, etc. When designing the bus network, the interests of both the operator and the passenger must be taken into account. Due to the conflicting nature of these interests, we treat the bus network design problem as a multicriteria decision-making problem. When designing the transit network, we try to maximize the number of satisfied passengers, to minimize the total number of transfers, and to minimize the total travel time of all served passengers.

In this paper we develop the model for the bus network design problem. Our approach is based on the Bee Colony Optimization (BCO) metaheuristics. The BCO algorithm is a stochastic, random-search technique that belongs to the class of population-based algorithms. This technique uses a similarity among the way in which bees in nature look for food, and the way in which optimization algorithms search for an optimum of a combinatorial optimization problem. The numerical experiments are performed on known benchmark problems, as well as on the problems generated by the authors of the paper. Our approach is competitive with

* Corresponding author. Tel.: +381 652106951.
E-mail address: duteodor@gmail.com (D. Teodorović).

other approaches in the literature, and it can generate high-quality solutions within negligible CPU times.

The paper is organized in the following way. Literature review is given in Section 2. Section 3 contains statement of the problem. Proposed solution to the problem is given in Section 4. The BCO approach to the transit network design problem is explained in details in Section 5. Experimental evaluation of the proposed approach is given in Section 6. Recommendations for future research and conclusion are given in Section 7.

2. Literature review

Various models for transit network design have been developed in the literature.

Lampkin and Saalmans (1967) proposed the first heuristic algorithm to design transit route network. In the first step, the proposed algorithm produces an initial skeleton route. In the next steps, the other nodes are inserting one by one into the skeleton route. The case study of a small town in the North of England is also presented in the paper. Silman, Barzily, and Passy (1974) proposed a two-staged approach for transit network design. They first generated a set of route-candidates through several iterations. The authors determined the optimal vehicle frequencies in the second stage. Silman et al. (1974) tried to minimize passengers travel time, while simultaneously taking care about the total number of passengers who cannot find seats. Byrne (1975) considered the case when the region served by the public transit is a segment of a circle and may be defined in polar coordinates. He proposed the model of a transit system that is built in polar coordinates with radial transit lines. Mandl (1979) proposed heuristic algorithm to find the set of the best transit routes. He reported the gained experiences in the case of some real world problems. Newell (1979) performed theoretical analysis of the bus route network design problem. He discussed various aspects of the problem and concluded that "in essence, our conclusion is that it would require a large computer and a vast amount of data to determine even a nearly optimal route geometry". Ceder and Wilson (1986) described the bus network design problem, analyzed previous concepts and proposed a two-level methodological approach for solving bus network design problem. Baaj and Mahmassani (1995) proposed route generation algorithm (RGA) for the design of transit networks. The proposed approach combined expert's knowledge and search techniques using Artificial Intelligence tools. Ceder and Israeli (1998) defined objective function that takes into account both passenger and operator interests. The proposed model for the transit network design problem combines mathematical programming, and decision-making techniques. When solving the bus route network design problem, Pattnaik, Mohan, and Tom (1998) proposed two step procedure. They generated the set of the route candidates in the first step. In the second step, the authors decided about the final set of routes by using the genetic algorithms. Bielli, Caramia, and Carotenuto (2002) applied genetic algorithm approach when considering bus network optimization problem. They tested their approach in the case of city of Parma, Italy. Chakroborty (2003) also proposed procedures for solving the urban transit network design problem based on the Genetic Algorithm. Lee and Vuchic (2005) considered the transit network design problem in the case of variable transit demand, under a given fixed total demand. The authors offered iterative approach that takes care about the relationship between variable transit trip demand and transit network design. The proposed approach is tested on the relatively small transit network. Guan, Yang, and Wirasinghe (2003) proposed the model for simultaneous optimization of transit line configuration and passenger line assignment. The proposed model is solved by branch and bound method. Fan and Machemehl (2006)

used the simulated annealing techniques to solve the optimal bus transit route network design problem. The proposed concept is tested in the case of three experimental networks. Zhao and Zeng (2006) combined genetic algorithm and simulated annealing while searching for the optimal route structures and headways. The authors tried to minimize transfers and total user cost, and to maximize service coverage. Zhao and Zeng (2007) developed the model for route network design, vehicle headways, and timetable assignment. The proposed approach combines simulated annealing, and tabu search. Desaulniers and Hickman (2007) reviewed the state-of-the-art models and approaches in solving complex public transit problems. Fan and Machemehl (2008) considered the design of public transportation networks in the case of variable demand. The authors developed multi-objective model. The solution methodology is based on Tabu search method. Guihaire and Hao (2008) classified 69 various approaches dealing with the transit network design and frequencies setting. They also indicated trends for future research. When solving route design and bus assignment problem, Pacheco, Alvarez, Casado, and Gonzalez-Velarde (2009) developed an algorithm based on local search strategy, as well as an algorithm based on a tabu search strategy. The authors showed the robustness of their approach with respect to variations in demand. The case study of the city of Burgos, Spain is presented in the paper. Mauttone and Urquhart (2009) developed Pair Insertion Algorithm (PIA) that can be used to generate initial solutions for a local improvement or evolutionary algorithm. The algorithm is inspired by the route generation algorithm (RGA) of Baaj and Mahmassani (1995). Kepaptsoglou and Karlaftis (2009) presented and reviewed research results in the area of transit route network design problem. Design objectives, operating environment parameters and solution approach are especially analyzed in the paper. Fan and Mumford (2010) proposed a model of the urban transit routing problem that evaluates candidate route sets. The proposed approach uses hill-climbing and simulated annealing techniques. Bagloee and Ceder (2011) studied the design a transit network for the actual-size road networks. The proposed algorithm was tested on the network of the city of Winnipeg, Canada, as well as on the transit network of Mandl benchmark. The review paper of Derrible and Kenneday (2011) is devoted to the applications of the graph theory in transit network design. Szeto and Wu (2011) studied the bus network design problem in the case of Tin Shui Wai, a suburban residential area in Hong Kong. The authors proposed the model that simultaneously performs the route design and bus frequency setting. The proposed solution method represents the combination of the genetic algorithm, and a neighborhood search heuristic. Miandoabchi, Farahani, Dullaert, and Szeto (2012) studied the design of urban road and public transit networks, The proposed multicriteria model decides about construction of new roads, adding lanes to the existing roads, lane allocation in two way streets, and the orientation of the one way streets. At the same time, the model proposes new routes of a given bus routes. Schoebel (2012) made the review of the various bus, railway, tram, and underground line planning models. Blum and Mathew (2012) studied the transit route network redesign problem. The proposed approach was tested in the case of city of Mumbai, India.

One can conclude that the majority of authors tried to minimize total travel time, or generalized cost. Simultaneously, greater part of papers introduced simplified assumption about fixed demand for transit services. More realistic assumption is the assumption that passenger flows depend on the transit network design, and that should be determined as a solution of an equilibrium problem. The decision variables are transit network route configuration and/or bus frequencies. Papers in the open literature also dealt with both type of passengers' assignment among possible transit routes: single path assignment and multiple path assignment. Due to the

problem hardness and computational complexity, the majority of papers offer heuristic, or metaheuristic approach.

3. Statement of the problem

Let us consider the road network shown in the Fig. 1. We denote this network by $G = (N,A)$, where N is the set of nodes, and A set of links (street segments). Nodes represent potential bus stops (intersections, zone centroids). Any path used by transit passengers is defined by a sequence of nodes, and links. We study the transit network design problem in the case of connected undirected street networks. Connected street network assumes that any two nodes in the network are connected by at least one path.

Within transit network design problem we search for the best possible set of routes R. In other words, we make the decision about the links from the set A to be included in the set of routes R, as well as the decision how to bring together chosen links into the fixed transit routes. In addition to this, authors in some cases also determine frequency of transit service on each of the defined routes.

The main indicator that we use to describe the level of transit service is the total travel time spent by the users of transit service. We express the quality of the solution generated in minutes. We calculate the total travel time of all passengers T in the network in the following way:

$$T = TT + w_1 TTR + w_2 TU \tag{1}$$

where

TT – total in-vehicle time of all served passengers,
TTR – the total number of transfers in the network,
TU – the total number of unsatisfied passengers (we assume that the passenger is unsatisfied when she/he has to make more than two transfers during the trip),
w_1 – time penalty for one transfer,
w_2 – time penalty for one unsatisfied passenger.

When measuring the quality of the solution generated, we take into account the total number of transfers, since transfers keep back passengers to use transit. Obviously, the total number of transfers may be decreased by optimizing the configuration of the transit network. In the same way, we use penalty for every unsatisfied passenger.

One potential solution of the transit network design problem is shown in Fig. 2.

We denote by d_{ij} the number of trips per time unit between node i and node j. We also denote by D the origin–destination matrix (O–D matrix):

$$D = \{d_{ij}|i,j \in [1,2,\ldots,|N|]\} \tag{2}$$

We denote by tr_{ij} the in-vehicle travel time between the node i and the node j. By TR we denote the travel time matrix:
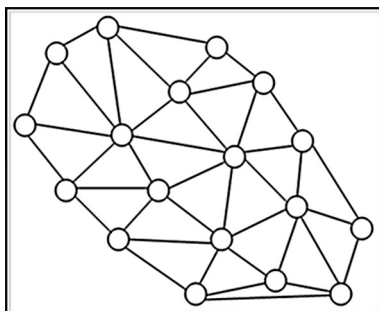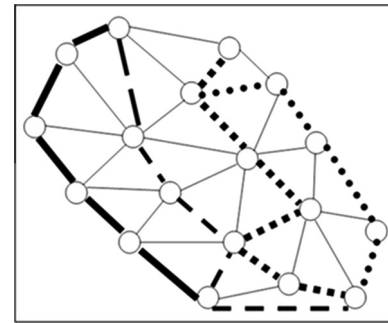

Fig. 1. Road network.

Fig. 2. Public transit network.

$$TR = \{tr_{ij}|i,j \in [1,2,\ldots|N|]\} \tag{3}$$

We assume in this paper that the network by $G = (N,A)$, the O–D matrix D, and the travel time matrix TR are given. We also assume that passengers choose the route based on the shortest travel time principle. Many researchers decomposed the transit network design problem into the following two stages: (1) generation of the set of transit routes; (2) determination of the frequency of service for each generated route. We do the same in this paper, and we focus exclusively on the generation of the set of transit routes.

The transit network design problem that we study in this paper could be defined in the following way: For a given set of n nodes, known origin–destination matrix D that describes demand among these nodes, and known travel time matrix TR, generate set of transit routes on a network in such a way to minimize the total travel time of all passengers T.

4. Proposed solution to the problem

The transit network design problem is one of the essential problems in the area of public transit. It has been shown that the network design problem is NP-hard problem (Magnanti & Wong, 1984). Due to its hardness and significance it has constantly been a test for researchers who approached it from various perspectives. As we indicated in the literature review, several heuristics and approaches based on meta-heuristics have been developed by different authors. The approach to the transit network design problem, that we propose, is based on the Bee Colony Optimization (BCO). The BCO represents one of the Swarm Intelligence techniques. Swarm Intelligence (Beni, 1988; Beni & Wang, 1989; Beni & Hackwood, 1992; Bonabeau, Dorigo, & Theraulaz, 1997) is the part of Artificial Intelligence based on study of actions of individuals in different decentralized systems. These decentralized systems are composed of physical individuals (robots, for example) or artificial ones that communicate among themselves, cooperate, collaborate, exchange information and knowledge and perform some tasks in their environment (Teodorović, 2008).

The Bee Colony Optimization (BCO) metaheuristic has been introduced by Lučić and Teodorović (2001, 2002, 2003a, 2003b). The BCO has been successfully applied to various engineering and management problems (Teodorović & Dell'Orco, 2005, 2008; Šelmić, Teodorović, & Vukadinović, 2010; Davidović, Ramljak, Šelmić, & Teodorović, 2011, 2012; Teodorović, Šelmić, & Mijatović-Teodorović, 2012; Todorović & Petrović, 2013). Nikolić and Teodorović (2013) recently performed empirical study of the BCO algorithm. The authors applied BCO to optimize numerous numerical test functions. They compared the obtained results with the results in the literature, and showed that the BCO is competitive with other methods.

The BCO is motivated by bees' behavior in the nature. The artificial bees utilize the principles used by honey bees for the period

of nectar collection process. The basic idea behind the BCO is to create the colony of artificial bees able to effectively solve difficult combinatorial optimization problems. Artificial bees explore through the search space, looking for the feasible solutions. Every artificial bee generates one solution to the problem. In order to discover good solutions, artificial bees collaborate and exchange information. By the use of collective knowledge and sharing information among themselves, artificial bees concentrate on more promising areas, and step by step reject solutions from the less promising ones. Artificial bees together generate and/or improve their solutions.

The algorithm consists of two alternating phases: *forward pass* and *backward pass*. In each forward pass, every artificial bee explores 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 again to the nest and start the second phase, the so-called backward pass. In the backward pass, all artificial bees share information about their solutions. In other words, bees make known the quality of the solution, i.e. the objective function value. Through the backward pass, every bee decides with a certain probability whether to abandon the created solution and become uncommitted follower, or dance and thus recruit the nestmates before returning to the created solution (bees with higher objective function value have greater chance to continue its own exploration). Every follower, choose a new solution from recruiters (Fig. 7) by the roulette wheel (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 could be of constructive or improving type. Constructive approach starts from scratch (Lučić & Teodorović, 2001, 2002, 2003a, 2003b; Teodorović, 2003, 2008, 2009; Šelmić et al., 2010; Davidović, Šelmić, & Teodorović, 2009, 2011, 2012; Dimitrijević, Teodorović, Simić, & Šelmić, 2011). Within this approach the analyst constructs a solution step by step. Quite the opposite, the improving approach begins from a complete solution. The complete solution (possible a feasible one) is typically generated randomly or by some heuristics. By perturbing that solution, we try to improve it (Davidović et al., 2011; Todorović & Petrović, 2013; Nikolić & Teodorović, 2013).

In this paper we apply the improvement version of the BCO algorithm. 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,
IT – the number of iteration,
NP – the number of forward and backward passes in a single iteration,
NC – the number of changes in one forward pass,
S – the best known solution.

The following is the pseudo code of the BCO algorithm:

procedure BCOi(**in** B,IT,NP,NC, **out** S)
  for i = 1 **to** B **do**
    Determine the initial solution for the bee i.
    Evaluate the solution of the bee i.
  S ← the best solution of the bees.
  for j = 1 **to** IT **do**
    for i = 1 **to** B **do**
      the bee i ← Set the initial solution.

  for k = 1 **to** NP **do**
    for i = 1 **to** B **do**
      for r = 1 **to** NC **do**
        Evaluate possible changes in the solution of the bee i.
        Taking into account evaluated values choose one change.
    for i = 1 **to** B **do**
      Evaluate solution of the bee i.
    for i = 1 **to** B **do**
      Make a decision whether the bee i is loyal.
    for i = 1 **to** B **do**
      if *the bee i is not loyal* **then**
        Chose one of the loyal bees that will be followed by the bee i.
    if *the best solution generated by the bees is* **better** *than the solution S* **then**
      S ← the best bee's solution.

5. The BCO approach to the transit network design problem

In this paper, we propose the BCO heuristic algorithm tailored for the transit network design problem. We propose the BCO algorithm that is based on the improvement concept. In other words, we first generate the initial feasible solution (the initial transit network). Then, artificial bees investigate solution space in the neighborhood of the current solution, and try to improve the solution. The modification of solution is performed through NP forward passes with in the single iteration. We assume that at the beginning of a network design, all artificial bees are in the hive. The hive is an artificial location, it is not connected either to bus lines.

5.1. Generating the initial solution

Let us consider bus line l whose terminals are located in the nodes i and j respectively (Fig. 3). Bus line l contains all nodes that belong to the shortest path between i and j. Let us denote with N_l the set of nodes connected by the line l.
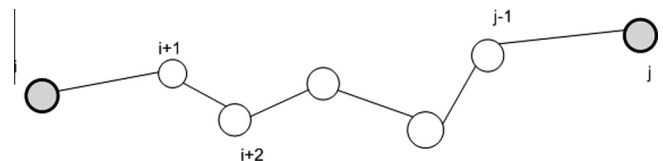

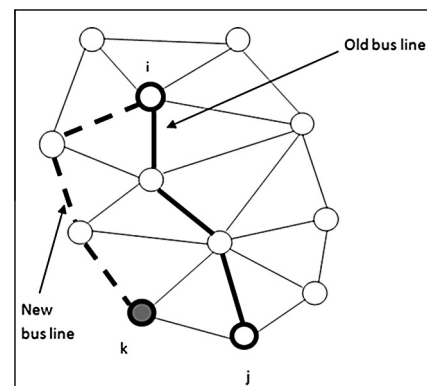**Fig. 3.** Bus line whose terminals are located in the nodes i and j.

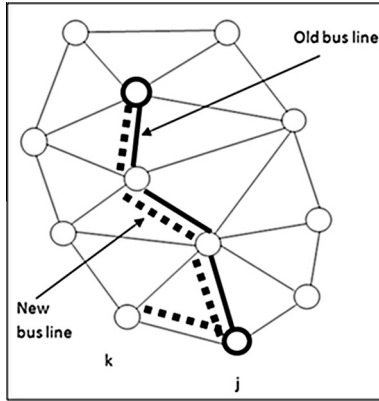**Fig. 4.** Solution modification by type 1 bee.

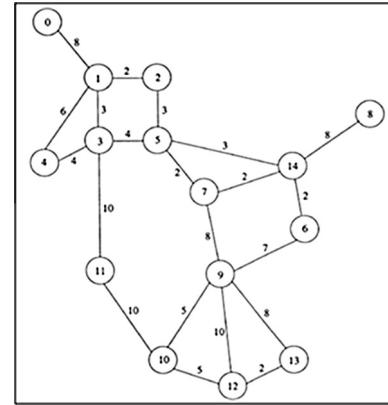Fig. 5. Solution modification by type 2 bee.

Fig. 9. Mandl's road network.
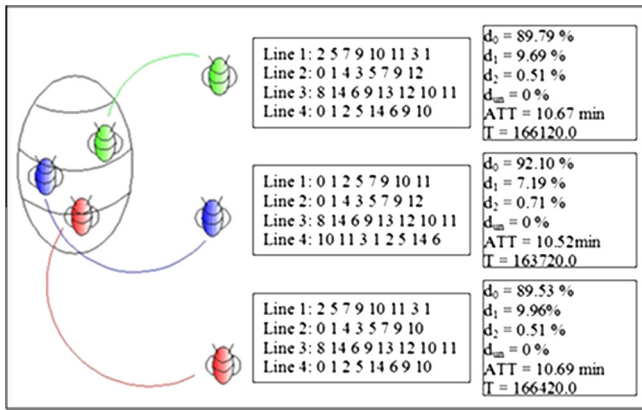
Fig. 6. Evaluation of the generated solutions.

This bus line could be used by the passengers that enjoy direct service, as well as by passengers that have to make the most two transfers during their trip. The total number of passengers ds_{ij} that enjoy the direct service along this bus line l equals:

$$ds_{ij} = \sum_{m \in N_l} \sum_{n \in N_l} d_{mn} \qquad (4)$$

We denote by DS the corresponding matrix that contains information about the number of passengers that enjoy the direct service:

$$DS = \{ds_{ij} | i, j \in [1, 2, \ldots, |N|]\} \qquad (5)$$

We propose a simple greedy algorithm to generate the initial solution. In this algorithm we try to connect, by the direct service, pairs of nodes that have high ds_{ij} values. In this way, we will increase the number of passengers that enjoy the direct service. The algorithm is composed of the following steps:

Fig. 7. Comparison of the generated solutions (Nikolić & Teodorović, 2013).

Fig. 8. Recruiting of uncommitted followers.

Table 1
The initial solution.

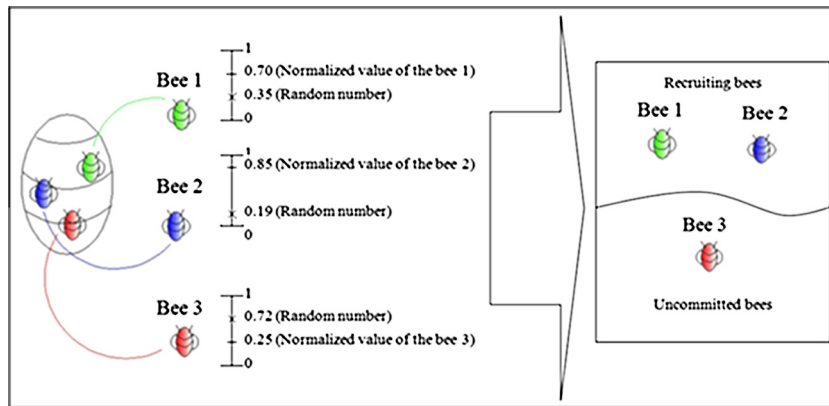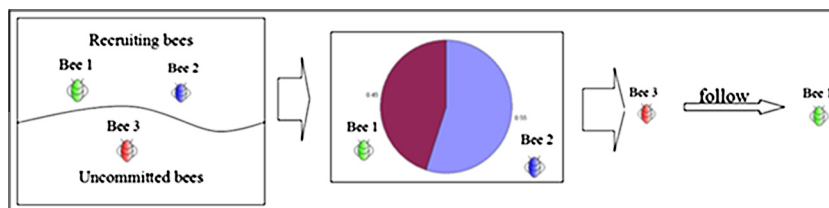| Situation	Number of routes	Route description
1	4	0 1 2 5 7 9 10
		4 3 5 7 9 12
		8 14 6 9 13
		0 1 2 5 14 6
2	6	0 1 2 5 7 9 10
		4 3 5 7 9 12
		8 14 6 9 13
		0 1 2 5 14 6
		9 10 11
		0 1 3 11
3	7	0 1 2 5 7 9 10
		4 3 5 7 9 12
		8 14 6 9 13
		0 1 2 5 14 6
		9 10 11
		0 1 3 11
		11 10 12 13
4	8	0 1 2 5 7 9 10
		4 3 5 7 9 12
		8 14 6 9 13
		0 1 2 5 14 6
		9 10 11
		0 1 3 11
		11 10 12 13
		0 1 4

Step 1: Prescribe the total number of bus lines *NBL* in the network. Denote the set of bus lines by *Y*. Set $Y = \phi$. Let $m = 1$.

Step 2: Find the pair of nodes that has the highest ds_{ij} value. Let this pair is the pair of nodes (a,b). The nodes a and b are the terminals of the new bus line. Find the shortest path between these two nodes. The nodes that belong to the shortest path represents stations in the bus lines. Add line *l* in set *Y*.

Step 3: Update the matrix *DS*, without taking into account passenger travel demands that is already satisfied.

Step 4: If $m = NBL$, stop; otherwise, set $m = m + 1$ and return to Step 2.

5.2. Solution modification

The major step in our algorithm is modification of the solution through *NP* forward passes within the single iteration.

In all our previous applications of the BCO algorithm, we used a set of homogeneous artificial bees. In this paper, for the first time, we operate with heterogeneous bees. We solve the transit network design problem by using two sets of artificial bees. The first set consists of bees type 1. The second set contains bees type 2. The type 1 artificial bees differ from the type 2 artificial bees only in the way in which they modify the solutions. When making decisions about the loyalty, as well as decisions to join some of the recruiters both bees of type1, and bees of type 2 behave in the same way.

5.2.1. Solution modification by type 1 bees

Let us note the given set of bus lines (Fig. 4). Type 1 bee chooses one line from the set of bus lines. Bee makes a decision which line to take according to probability. Probability of taking line *l* can be calculated as:

$$p_l = \frac{\frac{1}{ds_{ij}}}{\sum_{q \in L} \frac{1}{ds_{rs}}} \tag{6}$$

where:

i and j are the terminals of the line *l*,
L is the set of bus lines,
r and s are the terminals of the line *q*,
ds_{ij} – the total number of passengers that can travel without any transfer by using the line *l* that connects terminal *i* and terminal *j*.

Let us assume, for example, that type1 bee chose line whose terminals are *i* and *j*. In the next step, type 1 bee chooses one among two terminals. Let us assume that the type1 bee chose terminal *i*. Type 1 bee destroys selected bus line. New bus line will be created between terminal *i* and new terminal *k*, which could be chosen with the probability:

$$P_k = \frac{ds_{ik}}{\sum_{r \in N} ds_{ir}} \tag{7}$$

Table 2
Comparison between the initial solution obtained by the Greedy algorithm and the solutions obtained by the previous approaches.

Number of routes	Parameters	Mandl (1979)	Baaj and Mahmassani (1991)	Kidwai (1998)	Charkroborty and Dwivedi (2002)	Fan and Machemehl (2008)	Greedy algorithm (initial solution)
4	d_0	69.94	N	72.95	86.86	93.26	80.47
	d_1	29.93	N	26.92	12	6.74	12.33
	d_2	0.13	N	0.13	1.14	0	0.51
	d_{un}	0	N	0	0	0	6.68
	ATT	12.9	N	12.72	11.9	11.37	10.22
6	d_0	N	78.61	77.92	86.04	91.52	87.73
	d_1	N	21.39	19.68	13.96	8.48	11.75
	d_2	N	0	2.4	0	0	0.51
	d_{un}	N	0	0	0	0	0
	ATT	N	11.86	11.87	10.3	10.48	11.03
7	d_0	N	80.99	93.91	89.15	93.32	90.62
	d_1	N	19.01	6.09	10.85	6.36	8.86
	d_2	N	0	0	0	0.32	0.51
	d_{un}	N	0	0	0	0	0
	ATT	N	12.5	10.69	10.15	10.42	10.57
8	d_0	N	79.96	84.73	90.38	94.54	91.91
	d_1	N	20.04	15.27	9.62	5.46	7.58
	d_2	N	0	0	0	0	0.51
	d_{un}	N	0	0	0	0	0
	ATT	N	11.86	11.22	10.46	10.36	10.47

Table 3
The final solution obtained by the BCO algorithm.

Situation	Number of routes	Route description
1	4	0 1 2 5 7 9 10 11
		1 4 3 5 7 9 12 10
		8 14 6 9 7 5 3 11
		3 1 2 5 14 6 9 13
2	6	0 1 2 5 7 9 10 12
		0 1 4 3 5 7 9 10
		8 14 6 9 13 12 10 11
		0 1 2 5 14 6 9 10
		14 7 9 10 11 3 1 0
		8 14 5 2 1 4 3 11
3	7	0 1 2 5 7 9 13 12
		0 1 4 3 5 7 9 10
		8 14 6 9 13 12 10 11
		0 1 2 5 14 6 9 10
		5 7 9 10 11 3 1 0
		8 14 7 5 2 1 3 4
		6 14 7 5 3 11 10 12
4	8	0 1 2 5 7 9 10 12
		2 1 4 3 5 7 14 6
		8 14 6 9 10 11 3 5
		0 1 2 5 14 6 9 13
		8 14 5 2 1 3 11
		0 1 3 11 10 12 13 9
		1 4 3 5 7 9 10 12
		0 1 4 3 11 10 12 13

Let us assume that type 1 bee chose node k for the new terminal. Bee discovers the shortest path between the nodes i and k. This shortest path represents the new bus line (dashed line).

5.2.2. Solution modification by type 2 bees

Let us note the given set of bus lines (Fig. 5). Type 2 bee chooses one bus line in the same manner as the type 1 bee. After that, bee chooses in a random manner one, among two terminals, of this bus line. Let us assume, for example, that type 2 bee chose terminal j. In the next step, bee decides to destroy this terminal with the probability equals to P (The analyst prescribes P value). The new bus line generated in this way contains all old bus stops, except the de-

stroyed terminal. In this way, the old bus line was shortened. In the opposite case, if terminal survives (the terminals survives with the probability equal to $(1 − P)$), the old bus line will be expanded. The new bus line will contain one more bus stop. The new bus stop to be added to the bus line is chosen in a random manner from the nodes adjacent to the chosen terminal. In the case shown in Fig. 5, the node k was chosen to be included in the bus line, and the old bus line was expanded.

5.3. Bee's partial solutions comparison mechanism

Type 1 and type 2 bees return to the hive after modifying the solutions. All modified solutions are then evaluated by all bees (Fig. 6).

Let us denote by T_b $(b = 1, 2, \ldots, B)$ the total travel time of all passengers in the transit network generated by the bth bee. We denote by O_b normalized value of the total travel time T_b, i.e.:

$$O_b = \frac{T_{\max} - T_b}{T_{\max} - T_{\min}}, \quad b = 1, 2, \ldots, B \tag{8}$$

where T_{min} and T_{max} are respectively the smallest and the largest total travel time in all transit networks generated by all bees. The probability that bth bee (at the beginning of the new forward pass) is loyal to the previously generated solution is calculated in this paper in the following way:

$$p_b = e^{-(O_b - O_{\min})}, \quad b = 1, 2, \ldots, B \tag{9}$$

By using relation (9) and a random number generator, every type 1 and every type 2 artificial bee decides to become uncommitted follower, or to continue exploration from the generated solution (Fig. 7).

The superior the generated solution (higher O_b value), the higher the probability that the bee will be loyal to that solution.

5.4. Recruiting Process

In the case when at the beginning of a new stage bee does not want to start from the previously generated solution, the bee will go to the dancing area and will follow another bee (Fig. 8).

Table 4
The comparison between the final solution generated by the BCO approach and the previous approaches.

Number of routes	Parameters	Mandl (1979)	Baaj and Mahmassani (1991)	Kidwai (1998)	Charkroborty and Dwivedi (2002)	Fan and Machemehl (2008)	BCO algorithm
4	d_0	69.94	N	72.95	86.86	93.26	92.1
	d_1	29.93	N	26.92	12	6.74	7.19
	d_2	0.13	N	0.13	1.14	0	0.71
	d_{un}	0	N	0	0	0	0
	ATT	12.9	N	12.72	11.9	11.37	10.51
6	d_0	N	78.61	77.92	86.04	91.52	95.63
	d_1	N	21.39	19.68	13.96	8.48	4.37
	d_2	N	0	2.4	0	0	0
	d_{un}	N	0	0	0	0	0
	ATT	N	11.86	11.87	10.3	10.48	10.23
7	d_0	N	80.99	93.91	89.15	93.32	98.52
	d_1	N	19.01	6.09	10.85	6.36	1.48
	d_2	N	0	0	0	0.32	0
	d_{un}	N	0	0	0	0	0
	ATT	N	12.5	10.69	10.15	10.42	10.15
8	d_0	N	79.96	84.73	90.38	94.54	98.97
	d_1	N	20.04	15.27	9.62	5.46	1.03
	d_2	N	0	0	0	0	0
	d_{un}	N	0	0	0	0	0
	ATT	N	11.86	11.22	10.46	10.36	10.09

Table 5
Second example: bus lines generated by the Greedy algorithm (initial solution).

Line	Stations on the line
Line 1	59 48 92 52 23 2 45 40 56 74 50 6 94 68 12 26 57 21 104 18 79 101
Line 2	38 78 80 61 34 89 4 29 33 6 94 68 12 26 57 21 104 18 32 87
Line 3	58 55 51 41 36 28 45 40 56 74 50 6 94 68 12 26 57 21 104 18 73
Line 4	87 32 18 104 21 57 26 12 68 94 6 50 74 56 40 45 90 75 22 86 107
Line 5	24 37 13 44 4 29 56 40 45 28 72 3 88
Line 6	10 48 92 52 23 2 45 40 56 74 30 93 64 67 100
Line 7	54 53 13 44 4 29 33 6 94 68 12 26 57 21 104 18 79 101
Line 8	8 108 52 91 19 65 89 44 13 31 82 15
Line 9	82 5 105 95 33 6 94 68 12 26 35 102
Line 10	58 55 51 41 36 28 45 2 23 52 92 48 47 71
Line 11	0 70 26 12 68 94 6 50 74 56 40 45 2 23 108 8
Line 12	47 46 61 34 89 4 29 50 30 93 64 67
Line 13	58 55 51 41 36 28 45 40 56 29 4 89 76 60
Line 14	99 78 48 92 52 23 2 45 90 75 22 86 107
Line 15	17 9 68 94 6 50 74 56 40 45 2 23 52 92 48 25
Line 16	81 37 13 44 89 65 19 91 23 2 1 84
Line 17	87 32 18 104 21 57 26 12 68 94 6 50 74 56 40 45 28 72 3 88
Line 18	27 105 95 4 65 19 91 52 92 48 103
Line 19	15 5 105 95 29 56 40 45 90 75 22 86 107
Line 20	11 1 2 45 40 56 74 50 6 94 68 12 26 35 102
Line 21	7 49 68 94 6 50 74 56 40 45 2 23 52 92 48 103
Line 22	87 32 18 104 21 57 26 12 68 94 6 33 29 65 19 91
Line 23	10 59 46 76 44 13 31 5 27
Line 24	0 70 26 12 68 94 6 33 29 4 89 34 61 46 69
Line 25	38 10 48 92 52 23 2 45 28 72 3 88
Line 26	8 1 84 106 36 72 3 86 107
Line 27	54 53 13 44 4 29 56 40 45 90 75 22 86 107
Line 28	15 5 105 95 33 6 94 68 12 26 57 21 104 18 32 87
Line 29	16 9 68 94 6 33 29 4 89 34 61 46 71
Line 30	84 106 28 45 40 56 74 50 6 94 68 12 26 57 21 104 18 79 101
Line 31	24 37 13 44 4 29 33 6 94 68 12 26 57 21 104 18 73
Line 32	15 82 31 13 44 76 46 47 48 103
Line 33	98 64 7 12 26 57 21 104 18 79 101
Line 34	58 55 51 41 36 28 45 40 56 74 30 93 64 67 100
Line 35	62 1 2 45 40 56 74 50 6 94 68 12 26 57 21 104 18 32 87
Line 36	63 41 36 28 45 40 56 29 4 89 34 66
Line 37	60 34 61 46 47 48 92 11 62
Line 38	20 75 90 45 40 56 74 50 6 94 68 12 26 57 21 104 18 79 101
Line 39	15 5 105 95 33 50 30 93 64 67
Line 40	99 61 34 89 4 29 33 6 94 68 12 26 35 102
Line 41	39 75 90 45 2 23 52 92 48 59 69
Line 42	14 5 31 13 44 76 61 80 78 38
Line 43	100 67 64 93 30 74 56 40 45 90 75 22 86 107
Line 44	58 55 51 41 36 28 45 40 56 29 95 105 5 82
Line 45	47 46 61 34 89 4 29 33 6 94 68 12 26 57 21 104 18 79 101
Line 46	11 108 52 91 19 65 4 95 105 14
Line 47	43 9 68 94 6 33 29 4 44 13 97
Line 48	67 64 93 30 50 29 4 44 13 37 81
Line 49	77 30 94 68 12 26 57 21 104 18 79 101
Line 50	60 76 89 4 29 33 6 94 68 12 26 57 21 104 18 73
Line 51	54 53 13 44 89 65 19 91 52 108 62
Line 52	14 105 95 33 6 94 68 12 26 57 21 104 18 79 101
Line 53	54 53 13 44 4 29 56 40 45 28 36 41 51 55 58
Line 54	88 3 72 28 45 40 56 74 30 93 64 67 100
Line 55	11 1 84 106 36 41 51 55 58

Within the dance area the bee-dancers (recruiters) "advertise" different solutions. We assume that the probability the recruiter *b*'s solution will be chosen by any uncommitted bee equals:

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

where:

O_k – objective function value of the *k*th advertised solution;
R – the number of recruiters.

By using relation (10) and a random number generator, every uncommitted follower join one bee dancer (recruiter). Recruiters and recruited nestmates start exploration in the next forward pass from the solution discovered by the recruiter. All artificial bees are free to independently search the solution space and perform solution modification.

6. Experimental evaluation

The *BCO* approach that we propose is compared with other approaches that proved to be the best far in the literature. Comparison is performed in the case of Mandl's road network, as well as in the case of network that contains 110 Nodes and 275 links (http://people.brunel.ac.uk/~mastjjb/jeb/orlib/utraninfo.html).

In the proposed *BCO* algorithm we set *B* = 10, *NP* = 5, *NC* = 2. The values are chosen based on our prior experience. These values turn out to be both moderately small (resulting in rapid completion of each iteration) and sufficiently large to assure efficient knowledge

Table 6
Second example: bus lines generated by the BCO algorithm (final solution).

Line	Stations on the line
Line 1	80 78 46 99 69 38 10 25 48 92 52 23 2 45 40 56 74 50 6 94 68 12 26 57 21 104 18 79 101
Line 2	103 25 59 47 48 10 38 78 80 61 34 89 4 29 33 6 94 68 12 26 57 21 104 18 32 87 73 79 101
Line 3	58 55 51 41 36 28 45 40 56 74 50 6 94 68 12 26 57 21 104 18 73 87 32 79 101
Line 4	79 73 32 18 104 21 57 26 12 68 94 6 50 74 56 40 45 90 75 22 86 107
Line 5	54 53 81 24 37 13 44 4 29 56 40 45 28 72 3 39 20 88 75 36 106 84 1 62 11 108 52 91 83
Line 6	99 78 38 25 10 48 92 52 23 2 45 40 56 74 30 93 64 67 100 7 68 94 85 50 29 65 89 76 60
Line 7	19 65 109 95 96 105 5 97 37 81 54 53 13 44 4 29 33 6 94 68 12 26 57 21 104 18 79 32 87
Line 8	76 61 99 78 69 59 48 8 108 52 91 19 65 89 44 13 31 82 15 5 105 96 95 109 33 74 85 6 42
Line 9	54 37 97 82 5 105 95 33 6 94 68 12 26 35 102 0 70 57 16 17 49 100 9 7 67 64 98 30 85
Line 10	58 55 51 41 36 28 45 2 23 52 92 48 47 71 80 61 76 60 34 89 65 19 91 83 108 11 62 1 84
Line 11	98 64 67 100 49 7 9 17 43 16 57 102 0 70 26 12 68 94 6 50 74 56 40 45 2 23 108 92 8
Line 12	38 10 103 48 25 59 69 80 47 46 61 34 89 4 29 50 30 93 64 67 100 49 7 17 35 70 0 102 21
Line 13	58 55 51 41 36 28 45 40 56 29 4 89 76 60 34 66 61 80 99 46 71 47 48 10 103 25 59 69 78
Line 14	59 10 103 25 38 69 47 46 99 78 48 92 52 23 2 45 90 75 22 86 107
Line 15	35 57 16 43 17 9 68 94 6 50 74 56 40 45 2 23 52 92 48 25 38 69 47 71 46 76 44 13 53
Line 16	14 27 5 15 82 97 81 54 24 37 13 44 89 65 19 91 23 2 1 84 106 36 28 90 75 41 51 55 58
Line 17	73 87 32 18 104 21 57 26 12 68 94 6 50 74 56 40 45 28 72 3 88 86 22 75 36 106 84 1 8
Line 18	15 82 5 27 105 95 4 65 19 91 52 92 83 108 11 62 8 48 78 46 61 66 34 89 44 13 31 97 81
Line 19	10 59 46 61 34 60 76 44 13 37 24 53 54 81 97 82 15 5 105 95 29 56 40 45 90 75 22 86 107
Line 20	23 83 92 11 1 2 45 40 56 74 50 6 94 68 12 26 35 102 21 70 57 16 43 17 7 64 98 30 77
Line 21	43 17 26 57 21 0 70 102 35 12 9 7 49 68 94 6 50 74 56 40 45 2 23 52 92 48 103 10 38
Line 22	73 87 32 18 104 21 57 26 12 68 94 6 33 29 65 19 91 83 92 48 59 47 46 61 66 76 44 4 95
Line 23	66 61 80 78 48 25 103 10 59 46 76 44 13 31 5 27 105 96 95 4 109 65 19 91 52 23 108 11 62
Line 24	32 73 18 104 21 102 0 70 26 12 68 94 6 33 29 4 89 34 61 46 69 38 25 59 10 48 92 11 62
Line 25	82 5 105 96 95 109 65 89 34 61 80 71 69 59 25 38 10 48 92 52 23 2 45 28 72 3 88 75 20
Line 26	71 47 59 78 38 69 80 61 66 76 89 65 19 91 83 23 52 92 11 108 8 1 84 106 36 72 3 86 107
Line 27	5 82 31 97 81 24 37 54 53 13 44 4 29 56 40 45 90 75 22 88 20 39 72 28 36 41 3 86 107
Line 28	37 81 97 24 54 53 13 31 82 15 5 105 95 33 6 94 68 12 26 57 21 104 18 32 87 73 79 101
Line 29	49 17 26 70 35 12 43 16 9 68 94 6 33 29 4 89 34 61 46 71 80 78 59 25 48 92 108 52 83
Line 30	69 78 59 48 92 11 8 1 84 106 28 45 40 56 74 50 6 94 68 12 26 57 21 104 18 79 73 87 32
Line 31	54 53 81 24 37 13 44 4 29 33 6 94 68 12 26 57 21 104 18 73 87 32 79 101
Line 32	12 68 94 42 50 6 93 98 85 74 77 56 29 95 105 14 5 15 82 31 13 44 76 46 47 48 103 25 10
Line 33	48 47 46 76 89 65 109 33 74 50 30 94 42 6 93 85 98 64 7 12 26 57 21 104 18 79 73 32 87
Line 34	58 55 51 41 36 28 45 40 56 74 30 93 64 67 100 49 7 12 17 35 102 0 21 104 18 73 32 79 101
Line 35	31 13 44 4 65 19 91 52 92 8 62 1 2 45 40 56 74 50 6 94 68 12 26 57 21 104 18 32 87
Line 36	63 41 36 28 45 40 56 29 4 89 34 66 76 44 13 53 81 97 31 82 14 27 105 95 33 50 30 98 93
Line 37	23 91 19 65 89 44 76 34 61 46 47 48 92 11 62 8 1 2 45 28 36 75 3 51 63 41 72 39 20
Line 38	23 92 8 1 84 106 28 36 72 39 20 75 90 45 40 56 74 50 6 94 68 12 26 57 21 104 18 79 101
Line 39	54 81 53 24 97 82 15 5 105 95 33 50 30 93 64 67 100 9 12 43 16 57 21 104 18 73 87 32
Line 40	36 106 84 1 8 48 10 25 38 78 69 99 61 34 89 4 29 33 6 94 68 12 26 35 17 49 100 9 7
Line 41	11 62 1 84 106 36 72 41 63 51 3 39 75 90 45 2 23 52 92 48 59 69 78 99 46 76 44 13 97
Line 42	68 49 7 64 93 85 77 74 50 33 95 105 14 5 31 13 44 76 61 80 69 99 46 59 47 48 78 38 10
Line 43	101 79 73 87 32 18 104 21 57 26 17 9 100 67 64 93 30 74 56 40 45 90 75 22 86 107
Line 44	58 55 51 41 36 28 45 40 56 29 105 5 82 97 31 13 44 89 65 19 91 83 92 11 8 62 1 2
Line 45	69 99 78 48 10 103 25 59 47 46 61 34 89 4 29 33 6 94 68 12 26 57 21 104 18 79 32 87 73
Line 46	58 55 51 3 39 20 75 36 28 45 2 23 83 92 8 1 62 11 108 52 91 19 65 4 95 105 14 82 15
Line 47	12 17 43 9 68 94 6 33 29 4 44 13 97 82 14 27 105 96 95 109 65 19 91 83 92 48 103 10 38
Line 48	26 17 16 9 7 67 64 93 30 50 29 4 44 13 37 81 97 5 31 82 14 105 96 95 109 33 74 6 42
Line 49	24 37 53 81 97 5 27 105 96 95 29 109 33 42 50 77 30 94 68 12 26 57 21 104 18 79 32 73 87
Line 50	38 69 71 46 80 99 61 66 34 60 76 89 4 29 33 6 94 68 12 26 57 21 104 18 73 87 32 79 101
Line 51	5 15 82 97 37 81 54 53 13 44 89 65 19 91 52 108 11 1 84 106 36 72 39 75 20 3 51 55 58
Line 52	31 13 37 54 53 24 97 82 5 14 105 95 33 6 94 68 12 26 57 21 104 18 79 101
Line 53	95 96 105 14 27 5 97 37 54 53 13 44 4 29 56 40 45 28 36 41 51 3 86 22 88 39 72 75 90
Line 54	41 75 20 39 88 3 7 72 28 45 40 56 74 30 93 64 67 100 7 68 94 42 33 95 96 105 14 82 97 24
Line 55	27 14 82 15 5 105 95 4 44 76 46 59 48 8 11 1 84 106 36 41 51 55 58

exchange and discovering of a high quality final solutions. The penalty for unsatisfied passenger equals: average travel time + 50 min. The probability P that the red bee will destroy the terminal equals 0.2. The stopping criterion is the number of iteration $IT = 400$.

In the first experiment, we compare the BCO approach that we propose with the approaches proposed by Mandl (1979), Baaj and Mahmassani (1991), Kidwai (1998), Charkroborty and Dwivedi (2002) and Fan and Mumford (2010). The comparison is made in the case of Mandl's road network (Mandl, 1979) shown in Fig. 9.

We use the following parameters for comparison:

d_0 – the percentage of demand satisfied without any transfers,
d_1 – the percentage of demand satisfied with one transfer,
d_2 – the percentage of demand satisfied with two transfers,

d_{un} – the percentage of demand unsatisfied,
ATT – average travel time in minutes per transit user (mpu). This travel time includes transfer waiting times, and transfer time that is equal to 5 min per passenger.

This set of parameters was proposed by Fan and Mumford (2010). These authors performed comparison in four situations: 4 routes, 6 routes, 7 routes and 8 routes in each route set. We do the same in this paper. The results of comparison are given in Tables 1–4.

The initial solution (set of bus lines) generated by the greedy algorithm is shown in Table 1.

The comparison between the initial solution of the BCO approach and the previous approaches is given in Table 2.

One can see that simple greedy heuristic algorithm generated the initial solution whose characteristics are reasonable good. Artificial bee started to improve this initial solution through the search process. The final solution obtained by the BCO algorithm is given in Table 3.

The comparison between the final solution generated by the BCO approach and the previous approaches is given in Table 4.

The results obtained by the BCO approach have better values for d_0, and d_1 in 3 out of 4 cases. The BCO results have better values for ATT in 3 out of 4 cases.

The second experiment is performed for the bus network that contains 110 nodes and 275 links. The total number of daily trips equals 3,603,360. The task is to generate 55 bus lines. The total number of bus stops at any bus line cannot be bigger than 29.

The initial solution generated by the Greedy algorithm is given in Table 5.

The following are characteristics of the generated initial solution:

The total number of trips without transfer = 1,438,572 (d_0 = 39.92%).
The total number of trips with one transfer = 1,607,740 (d_1 = 44.62%).
The total number of trips with two transfers = 224,480 (d_2 = 6.23%).
The total number of unsatisfied passengers = 332,568 (dun = 9.23%).
The average travel time in minutes per transit user ATT = 34.90 min.

As in the previous example, BCO algorithm significantly improved the initial solution. The final solution generated by the BCO algorithm is given in Table 6.

The following are characteristics of the generated initial solution:

d_0 = 59.65%,
d_1 = 40.10%,
d_2 = 0.25%,
d_{un} = 0%,
ATT = 36.16 min.

The obtained average travel time *ATT* (in minutes per transit user) is composed of the following two components:

$$36.16 = 34.13 + 2.03$$

The first component represents the average travel time, while the second component represents the average penalty per transit user caused by transfer. In order to better estimate the quality of the obtained final solution we calculated the average travel time per passenger in the case when every passenger exclusively follow the shortest path when traveling between the origin and destination. This average travel time equals 33.84 min. No one transit network cannot have average travel time per transit user less than 33.84 min. This value represents lower bound of the average travel time. The ATT value obtained by the BCO algorithm is relatively close to the lower bound, so one can conclude that the quality of the solution generated by the BCO algorithm is relatively high.

7. Conclusion

We developed the model for the transit network design problem. The transit network design problem is a large combinatorial problem whose optimal solution is difficult to find, therefore a heuristic approach must be used. The model proposed in this paper is based on the Swarm Intelligence concepts. We tried to maximize the number of served passengers, to minimize the total in-vehicle time of all served passengers, and to minimize the total number of transfers in the network. We clearly showed that the proposed BCO algorithm is competitive with other approaches in the literature, and that it can generate high-quality solutions within reasonable CPU times.

The challenge for the future research is to test the offered BCO concept simultaneously with the assumption that passenger flows depend on the transit network design, as well as in the case of the multiple path passengers' assignment among possible transit routes.

The proposed concept should be expanded in the future research by including in the analysis available number of vehicles, as well as some other operational constraints.

References

Baaj, M. H., & Mahmassani, H. (1991). An AI-based approach for transit route system planning and design. *Journal of Advanced Transportation, 25*(2), 187–210.

Baaj, M. H., & Mahmassani, H. S. (1995). Hybrid route generation heuristic algorithm for the design of transit networks. *Transportation Research C, 3*(1), 31–50.

Bagloee, S. A., & Ceder, A. (2011). Transit-network design methodology for actual-size road networks. *Transportation Research B, 45*, 1787–1804.

Beni, G. (1988). The concept of cellular robotic system. In *Proceedings of the 1988 IEEE international symposium on intelligent control* (pp. 57–62). Los Alamitos, CA: IEEE Computer Society Press.

Beni, G., & Hackwood, S. (1992). Stationary waves in cyclic swarms. In *Proceedings of the 1992 international symposium on intelligent control* (pp. 234–242). Los Alamitos, CA: IEEE Computer Society Press.

Beni, G., & Wang, J. (1989). Swarm intelligence. In *Proceedings of the seventh annual meeting of the robotics society of Japan* (pp. 425–428). Tokyo: RSJ Press.

Bielli, M., Caramia, M., & Carotenuto, P. (2002). Genetic algorithms in bus network optimization. *Transportation Research Part C: Emerging Technologies, 10*(1), 19–34.

Blum, J. J., & Mathew, T. V. (2012). Implications of the computational complexity of transit route network redesign for metaheuristic optimisation systems. *IET Intelligent Transport Systems, 6*, 124–131.

Bonabeau, E., Dorigo, M., & Theraulaz, G. (1997). *Swarm intelligence*. Oxford University Press.

Byrne, B. F. (1975). Public transportation line positions and headways for minimum user and system cost in a radial case. *Transportation Research, 9*, 97–102.

Ceder, A., & Israeli, Y. (1998). User and operator perspectives in transit network design. *Transportation Research Record, 1623*, 3–7.

Ceder, A., & Wilson, H. M. (1986). Bus network design. *Transportation Research, 20B*(4), 331–344.

Chakroborty, P. (2003). Genetic algorithms for optimal urban transit network design. *Computer-Aided Civil and Infrastructure Engineering, 18*, 184–200.

Charkroborty, P., & Dwivedi, T. (2002). Optimal route network design for transit systems using genetic algorithms. *Engineering Optimization, 34*(1), 83–100.

Davidović, T., Šelmić, M., & Teodorović, D. (2009). Scheduling independent tasks: Bee colony optimization approach. In *Proceedings of the 17th Mediterranean conference on control and automation, MED'09, Thessaloniki, Greece, June 24–26* (pp. 1020–1025).

Davidović, T., Ramljak, D., Šelmić, M., & Teodorović, D. (2011). Bee colony optimization for the p-center problem. *Computers & Operations Research, 38*, 1367–1376.

Davidović, T., Šelmić, M., Teodorović, D., & Ramljak, D. (2012). Bee colony optimization for scheduling independent tasks to identical processors. *Journal of Heuristics, 18*(4), 549–569.

Derrible, S., & Kenneday, C. (2011). Applications of graph theory and network science to transit network design. *Transport reviews, 31*, 495–519.

Desaulniers, G., & Hickman, M. D. (2007). Public transit. In C. Barnhart, & G. Laporte (Eds.), *Vol. 14. Handbook in OR & MS* (pp. 69–127).

Dimitrijević, B., Teodorović, D., Simić, V., & Šelmić, M. (2011). A bee colony optimization approach to solving the anti-covering location problem. *Journal of Computing in Civil Engineering, 26*(6), 759–768.

Fan, W., & Machemehl, R. B. (2006). Optimal transit route network design problem with variable transit demand: Genetic algorithm approach. *Journal of Transportation Engineering – ASCE, 132*, 40–51.

Fan, W., & Machemehl, R. B. (2008). Tabu search strategies for the public transportation network optimizations with variable transit demand. *Computer Aided Civil and Infrastructure Engineering, 23*, 502–520.

Fan, L., & Mumford, C. L. (2010). A metaheuristic approach to the urban transit routing problem. *Journal of Heuristics, 16*, 353–372.

Guan, J. F., Yang, H., & Wirasinghe, S. C. (2003). Simultaneous optimization of transit line configuration and passenger line assignment. *Transportation Research Part B, 40*(10), 885–902.

Guihaire, V., & Hao, J. K. (2008). Transit network design and scheduling: A global review. *Transportation Research A, 42*, 1251–1273.

<http://people.brunel.ac.uk/~mastjjb/jeb/orlib/utraninfo.html>.

Kepaptsoglou, K., & Karlaftis, M. (2009). Transit route network design problem: Review. *Journal of Transportation Engineering – ASCE, 135*(8), 491–505.

Kidwai, F. A. (1998). Optimal design of bus transit network: A genetic algorithm based approach. Phd. dissertation, Indian Institute of Technology, Kanpur, India.

Lampkin & Saalmans (1967). The design of routes service frequencies and schedules for municipal bus undertaking: A case studys. *Operational Research Quarterly, 18*(4), 375–397.

Lee, Y. J., & Vuchic, V. R. (2005). Transit network design with variable demand. *Journal of Transportation Engineering – ASCE, 131*, 1–10.

Lučić, P., & Teodorović, D. (2001). Bee system: Modeling combinatorial optimization transportation engineering problems by swarm intelligence. In *Preprints of the TRISTAN IV triennial symposium on transportation analysis, Sao Miguel, Azores Islands, Portugal* (pp. 441–445).

Lučić, P., & Teodorović, D. (2002). Transportation modeling: An artificial life approach. In *Proceedings of the 14th IEEE "international conference on tools with artificial intelligence", Washington, DC* (pp. 216–223).

Lučić, P., & Teodorović, D. (2003a). Computing with bees: attacking complex transportation engineering problems. *International Journal on Artificial Intelligence Tools, 12*, 375–394.

Lučić, P., & Teodorović, D. (2003b). Vehicle routing problem with uncertain demand at nodes: The bee system and fuzzy logic approach. In J. L. Verdegay (Ed.), *Fuzzy sets in optimization* (pp. 67–82). Heidelberg, Berlin: Springer-Verlag.

Magnanti, T. L., & Wong, R. T. (1984). Network design and transportation planning: Models and algorithms. *Transportation Science, 18*(1), 1–55.

Mandl, C. E. (1979). Evaluation and optimization of urban public transportation network. *European Journal of Operational Research, 5*, 396–404.

Mauttone, A., & Urquhart, M. E. (2009). A route set construction algorithm for the transit network design problem. *Computers & Operations Research, 36*, 2440–2449.

Miandoabchi, E., Farahani, R. Z., Dullaert, W., & Szeto, W. Y. (2012). Hybrid evolutionary metaheuristics for concurrent multi-objective design of urban road and public transit networks. *Networks & Spatial Economics, 12*, 441–480.

Newell, G. (1979). Some issues relating to the optimal design of bus routes. *Transportation Science, 13*, 20–35.

Nikolić, M., & Teodorović, D. (2013). Empirical study of the bee colony optimization (BCO) algorithm. *Expert Systems with Applications, 40*(11), 4609–4620.

Pacheco, J., Alvarez, A., Casado, S., & Gonzalez-Velarde, J. L. (2009). A tabu search approach to an urban transport problem in northern Spain. *Computers & Operations Research, 36*, 967–979.

Pattnaik, S., Mohan, S., & Tom, V. (1998). Urban bus transit route network design using genetic algorithm. *Journal of Transportation Engineering, 124*(4), 368–375.

Schoebel, A. (2012). Line planning in public transportation: Models and methods. *OR SPECTRUM, 34*, 491–510.

Šelmić, M., Teodorović, D., & Vukadinović, K. (2010). Locating inspection facilities in traffic networks: An artificial intelligence approach. *Transportation Planning and Technology, 33*, 481–493.

Silman, L. A., Barzily, Z., & Passy, U. (1974). Planning the route system for urban buses. *Computers & Operations Research, 1*, 201–211.

Szeto, W. V., & Wu, Y. (2011). A simultaneous bus route design and frequency setting problem for Tin Shui Wai, Hong Kong. *European Journal of Operational Research, 209*, 141–155.

Teodorović, D. (2003). Transport modeling by multi-agent systems: A swarm intelligence approach. *Transportation Planning and Technology, 26*, 289–312.

Teodorović, D. (2008). Swarm intelligence systems for transportation engineering: Principles and applications. *Transportation Research Part C: Emerging Technologies, 16*, 651–782.

Teodorović, D. (2009). Bee colony optimization (BCO). In C. P. Lim, L. C. Jain, & S. Dehuri (Eds.), *Innovations in swarm intelligence* (pp. 39–60). Berlin, Heidelberg: Springer-Verlag.

Teodorović, D., & Dell'Orco, M. (2005). Bee colony optimization – a cooperative learning approach to complex transportation problems. In *Advanced OR and AI methods in transportation. Proceedings of the 10th meeting of the EURO working group on transportation, Poznan, Poland* (pp. 51–60).

Teodorović, D., & Dell'Orco, M. (2008). Mitigating traffic congestion: solving the ride-matching problem by bee colony optimization. *Transportation Planning and Technology, 31*, 135–152.

Teodorović, D., Šelmić, M., & Mijatović-Teodorović, Lj. (2012). Combining case-based reasoning with bee colony optimization for dose planning in well differentiated thyroid cancer treatment. *Expert Systems with Applications, 40*(6), 2147–2155.

Todorović, N., & Petrović, S. (2013). Bee colony optimization for nurse rostering. *IEEE Transactions on Systems Man & Cybernetics: Systems, 43*(2), 467–473.

Zhao, F., & Zeng, X. (2006). Simulated annealing-genetic algorithm for transit network optimization. *Journal of Computing in Civil Engineering, 20*(1), 57–68.

Zhao, F., & Zeng, X. G. (2007). Optimization of transit route network, vehicle headways and timetables for large-scale transit networks. *European Journal of Operational Research, 186*, 841–855.