



A Differential Evolution for Optimization of Multiobjective Urban Transit Routing Problem

¹Ahmed Tarajo BUBA and ²Lai Soon LEE

^{1,2}Department of Mathematics, Faculty of Science,
Universiti Putra Malaysia, 43400 UPM Serdang, Selangor, Malaysia.

²Laboratory of Computational Statistics and Operations Research, Institute for Mathematical Research,
Universiti Putra Malaysia, 43400 UPM Serdang, Selangor, Malaysia.
Corresponding Author Email: abtarajo72@yahoo.com

Abstract

In this paper, the urban transit routing problem is addressed by using a real-world urban transit network. Given the road network infrastructure and the demand, the problem consists in designing routes such that the service level as well as the operator cost are optimized. The optimality of the service level is measured in terms of average journey time and the route set length. A differential evolution approach is proposed to solve the problem. An improved sub-route reversal repair mechanism is introduced to deal with the infeasibility of route sets. Computational results on a real network produce solutions that are close to the lower bound values of the passenger and the operator costs. In addition, the proposed algorithm produces approximate Pareto fronts that enable the transit operator to evaluate the trade-off between the passenger and passenger costs.

Keywords: transit network design; differential evolution; repair mechanism; urban routing.

1. Introduction

Over the years, the public transportation system plays a significant role in daily lives of people in many cities of the world. With the rise in population and urbanization of many cities, especially in developing and emerging countries, there are significant increase of usage of private vehicles for daily commuting. These issues have contributed to problems like traffic congestion and accidents, unreliable travel times, pollution, etc. One of the solutions to handle these problems is to improve the public transportation systems through the proper design of transit networks that can accommodate the huge travel demands at different levels of operations. Due to its complex transit travel time characteristics, which include in-vehicle travel time, waiting time, transfer time, and transfer penalties, it has been a difficult task to optimize transit networks [19]. This work is restricted to a system based on buses, which is rarely tackled by transport planners. In most cases, bus networks evolve incrementally; new services are being added as the city develops. Over the decades, many public transit networks in many cities have not been reappraised from anywhere between 20 to 50 years [4].

Several studies in the literature attempted to optimize the passenger cost, operator cost, or both for the Urban Transit Routing Problem (UTRP) (see, [11], [21], [9], and [7]). The passenger costs are total travel time (i.e. in-vehicle and transfer penalty), average travel time, and goodness of route sets. Operator cost is designated by total route set length which accounts for the mileage, vehicle operation hours, fuel consumption, and driver schedule. Comprehensive reviews of the previous work on UTRP are provided in [13], [17], [12], [14], and [6]. Based on the literature, heuristic and metaheuristic approaches

are dominantly used for solving the UTRP (see, [25], [8], [23], [11], [3], [24], [16], [22], [18], [27], and [1]).

2. Material and Methods

2.1 Urban Transit Routing Problem

The UTRP involves the development of transit routes on an existing road network with associated link travel times and predefined demand (stop) points, such that the routes optimally satisfy some user-defined objectives, subject to the constraints. Generally, passengers would prefer to travel to their destination within the shortest time possible, but avoiding the discomfort associated with too many transfers. The passenger cost for a route set, \mathfrak{R} is defined as the average journey time over all passengers, which consists of in-vehicle travel time plus transfer penalty. On the other hand, the operator costs depend on many factors including the fleet size required to maintain the needed service level, the daily distance covered by the vehicles, vehicle operating hours and the cost of employing enough drivers.

The UTRP can be formally defined as (see, [15]): Given a road network represented as a graph $G = (V, E)$, where $V = \{v_1, \dots, v_n\}$ is a set of vertices representing demand points (bus stops) and $E = \{e_1, \dots, e_m\}$ a set of edges representing street segments. The weight for each edge, W_{e_i} , defines the time it takes to traverse edge e_i , and matrix $D_{n \times n}$ such that D_{v_i, v_j} gives the passenger demand between a pair of vertices v_i and v_j .

Let $G_R = (V_R, E_R)$ be the subgraph formed by a route R_i , where R_i is defined as a simple path (i.e. no loops/repeated vertices)

through the graph G . A solution is defined as a set of overlapping routes

$\mathfrak{R} = \{R_1, \dots, R_r\}$ where r , is the number of routes in the route set, prespecified by the service provider. The length of R_i is measured by the minimum (m_1) and maximum (m_2) number of vertices for service quality. Additionally, Let $\tau_{i,j}(\mathfrak{R})$ denote the shortest journey time from any pair of vertices (v_i, v_j) using route set \mathfrak{R} calculated using Dijkstra's algorithm [10], which consist of in-

$$\min C_p(\mathfrak{R}) = \frac{\sum_{i,j=1}^n D_{ij} \tau_{ij}(\mathfrak{R})}{\sum_{i,j=1}^n D_{i,j=1}}, \quad (1)$$

$$C_o(\mathfrak{R}) = \sum_{\forall R_i \in \mathfrak{R}} \sum_{\forall e_j \in R_i} W_{e_j}, \quad (2)$$

Eq. 1 is the passenger cost (C_p) for a route set \mathfrak{R} , defined as the average journey time over all passengers. Eq. 2 is the operator cost (C_p) defined by the total route set length. Constraint (3) specifies that each route should be contained between m_1 and m_2 vertices. Constraint (4) ensures that the solution contains the correct number of routes. Constraint (5) ensures that all vertices in V are in at least one route in \mathfrak{R} , and constraint (6) specifies that a path exists between all pairs of vertices in G_R . If Constraint (6) is satisfied, then $G_R = (V, \cup_{i=1}^{|\mathfrak{R}|} E_{R_i})$. For this problem formulation, the following assumptions are also made (see, [7]):

- i. Each route in the route set is free from repeated nodes. Hence, no cycles or backtracks should be allowed in the individual routes.
- ii. All nodes must be included in the route set to form a complete route set.
- iii. The demand, travel time, and distance matrices are symmetrical along the same route.
- iv. The demand level is inelastic throughout the period of the study and passenger choice of routes is based on the shortest travel time.
- v. The policy headway is relaxed. It is assumed there are adequate vehicles and capacity, and total travel time consist only of in-vehicle travel time plus transfer penalties at five minutes for each transfer.

2.2 Differential Evolution

Differential evolution (DE) is a variant of evolutionary algorithms proposed by [26] for continuous optimization problems. DE has some similarity to genetic algorithms due to its use of crossover, mutation and selection operators. However, DE depends heavily on mutation as the primary search mechanism. During the

vehicle travel and transfer penalty. Therefore, for a given origin-destination matrix $D_{n \times n}$ that represents demand among these vertices together with a travel time matrix TR , where tr_{ij} is the in-vehicle travel time between vertices i and j . The objective of the multiobjective UTRP is to determine a set of transit route networks that minimizes the Eq. 1 and Eq. 2 while meeting all the constraints of Eq. 3 – 6.

$$\text{subject to } m_1 \leq |V_{R_i}| \leq m_2 \quad \forall R_i \in \mathfrak{R}, \quad (3)$$

$$|\mathfrak{R}| = r, \quad (4)$$

$$G_R = (\cup_{i=1}^{|\mathfrak{R}|} V_{R_i}, \cup_{i=1}^{|\mathfrak{R}|} E_{R_i}) \text{ is connected}, \quad (5)$$

$$\cup_{i=1}^{|\mathfrak{R}|} V_{R_i} = V. \quad (6)$$

implementation, the DE adapts the search step automatically through the mutation to achieve the best value.

The proposed DE framework consists of two stages. The construction heuristic algorithm proposed by [21] is used to generate the initial population of vectors in the first stage, followed by a DE algorithm to determine the optimal route networks in the second stage. During the implementation of the framework, it is likely that some infeasible vectors could be generated. Some studies considered such vectors to be rejected and the whole initialization procedure together with feasibility checks are then repeated to construct new feasible vectors (see, [21], [9], and [1]). In this study, we proposed an improved version of the sub-route reversal (SRR) repair mechanism from our previous work to deal with the infeasible vectors. The improved SRR is executed as in the steps outlined in SRR, detailed in [7], except that in **STEP 5, Case 1** (and **Case 2**), if any of the nodes from the sub-route of s_2 (s_1 for **Case 2**) cannot be appended to the reversed sub-route of s_1 (s_2 for **Case 2**), it is returned to the missing node list in **STEP 1**, provided such node(s) do not appear in any routes of the current infeasible route set. This process of updating the missing node list is repeated for a predefined number of iterations, k .

Each individual is a complete route set known as a vector in DE terminology, from a given road network. In the proposed DE, the route set is represented as a vector of integers with the nodes listed in the order in which there are visited and each route is separated by an “*”. A sample vector containing four routes is shown in Fig. 1, where the first route is visited in the sequence of nodes 0, 1, 3, and 4. The length of a route is measured by the number of nodes it contains. Note that the population size, N_p is kept constant during the execution of the DE algorithm.

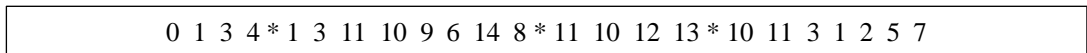


Figure 1: A Sample Route Set (vector) with 4 Routes

A mutation operator helps in maintaining the diversity of the population. An identical-point mutation operator based on the modified version of route crossover in [23] is used to handle the discrete nature of UTRP. In this mutation, a random node that appears in two routes of a vector is chosen. The substrings before the random node in the chosen routes swap their position between the two routes to create a new *Noisy Random* Vector. If the new vector is infeasible, the repair mechanism will be performed. If the repair mechanism failed to repair the infeasible vector, another vector is randomly selected to perform the mutation.

As for the crossover operator, the uniform route crossover is utilized [5]. In this crossover, a pair of *Trial* vectors is generated through selecting the vector components values either from the *Target* or the *Noisy Random* vector using a crossover mask where each route in the *Trial* vector is constructed by copying the route either from the *Target* or the *Noisy Random* vector. In order to reduce the possibility of getting an infeasible result, the mask should not be highly diverse. However, it is necessary to check for feasibility of the *Trial* vectors.

To construct the new population for the next generation, elitism selection is used, where the best vector between the *Target* and the

Trial vectors will be selected for the next generation. This ensures the average fitness of the population does not deteriorate. The detailed steps of the proposed DE for solving the multiobjective

UTRP can be found in [7]. The framework of the proposed DE is shown in **Algorithm 1**.

Algorithm 1: Differential Evolution for Multiobjective Urban Transit Routing Problem [7]

```

1: Generate  $N_p$  candidate route set based on heuristic in [21] with improved SRR repair mechanism
2: for  $i := 1$  to  $N_p$ 
3:   fitness evaluation
4: end for
5: for  $n := 1$  to  $G$ 
6:   for  $i := 1$  to  $N_p$ 
7:     set Target vector =  $X_{i,n}$ 
8:     select randomly a vector (except the selected Target vector,  $X_{i,n}$ ) in the population
9:     apply identical point mutation to generate a Noisy Random vector,  $V_{i,n}$  (repair if infeasible)
10:    apply uniform crossover between  $X_{i,n}$  and  $V_{i,n}$  to generate a pair of Trial vectors,  $U_{i,n}$  (repair if infeasible)
11:    fitness evaluation of  $U_{i,n}$ 
12:    elitism selection
13:    if Trial vector fitness  $\leq$  Target vector fitness
14:      new_population [ $i$ ] = Trial vector,  $U_{i,n}$ 
15:    else
16:      new_population [ $i$ ] = Target vector,  $X_{i,n}$ 
17:    end for
18:     $N_p =$  new_population
19: end for
20: return BEST

```

3. Results

The proposed DE algorithm is implemented on a real-life network (Abuja Transit operator) in Nigeria. It is a small but dense city with 30 nodes, 44 bidirectional links, and 15 routes with many overlapping routes (see, Fig. 2). During the peak hour, the transit demand is composed by 422,186 units with the highest node pair travel demand at 4800 units. Both travel time and travel demand matrices have a “many-to-many” structure. Each route is made up of a minimum of two nodes and a maximum of 15 nodes with a five minute penalty for each transfer.

While treating the multiobjective UTRP, [21] proposed a flexible approach formulation of this problem without scalarizing the objective functions that aim to determine a Pareto frontier for the analysis of the trade-off between the costs of the passenger and the operator. Inspired by [9], the implementation of the proposed DE

for multiobjective UTRP consists of alternating the objective only when the entire population of the first objective (C_p) has converged. Therefore, the proposed algorithm will only switch once and both of the objectives will start with the same initial population that has been recorded earlier. The proposed algorithm is coded in Python 2.7.6.4 and executed on a computer with 1.60 GHz Intel Core™i5-4200 CPU and 4.00 GB of RAM. A population of 30 vectors and 200 generations is used for the computation while the algorithm will terminate earlier when no improvement is observed over 50 consecutive generations. The proposed algorithm is performed for 10 runs and the best result is reported in Table 1. The best route sets constructed by the proposed algorithm are given in Tables 2 and 3. The approximate Pareto front achieved by the proposed DE for the real-life network is shown in Fig. 3 so that the decision maker can evaluate the best suited solution.

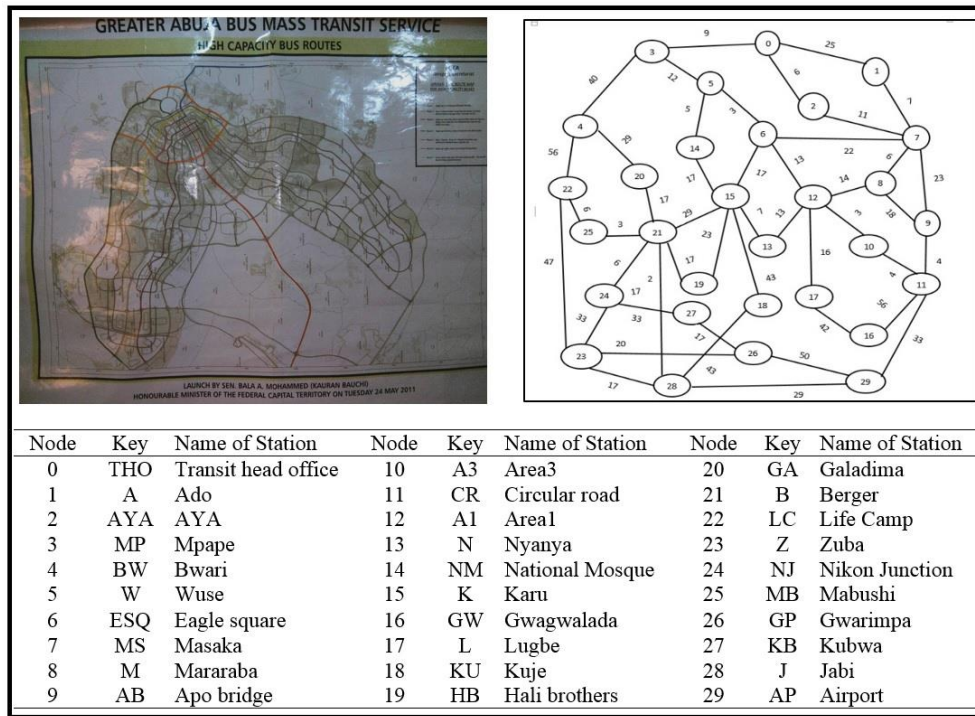


Figure 2: Nigeria Real Transit Network

Table 1: Best Results (15 routes) of Nigeria Network

Parameters	Proposed DE algorithm	
	Passenger	Operator
d_0	40.11	17.12
d_1	45.08	26.60
d_2	11.78	22.84
d_3	3.03	17.67
d_4	0.00	10.27
d_5	0.00	3.19
d_{un}	0.00	2.31
C_p	42.88(38.90)	60.83
C_o	3891	569 (465)

Table 2: Best Route Sets (for Passenger) of Nigeria Network

Routes	Sequence of Routes
1	27 – 26 – 29 – 11 – 10 – 12 – 8 – 7 – 2 – 6 – 5 – 3 – 4 – 20
2	6 – 2 – 0 – 3 – 21 – 24 – 23 – 28 – 29 – 11 – 9 – 8 – 7 – 1
3	7 – 6 – 2 – 8 – 12 – 17 – 16 – 11 – 29 – 26 – 23 – 24 – 27 – 25 – 22
4	19 – 21 – 28 – 18 – 15 – 6 – 7 – 2 – 0 – 3 – 4 – 22 – 25 – 27 – 24
5	13 – 15 – 14 – 5 – 3 – 21 – 25 – 22 – 23 – 26 – 29 – 11 – 16 – 17 – 12
6	7 – 1 – 0 – 3 – 21 – 20 – 4 – 22 – 23 – 28 – 29 – 11 – 9 – 8 – 2
7	16 – 17 – 12 – 8 – 2 – 6 – 15 – 19 – 21 – 20 – 4 – 22 – 25 – 21
8	17 – 12 – 8 – 9 – 11 – 29 – 26 – 23 – 22 – 25 – 21 – 19 – 15 – 6 – 5
9	1 – 0 – 2 – 8 – 7 – 6 – 15 – 14 – 5 – 3 – 4 – 20 – 21 – 25 – 22
10	28 – 21 – 19 – 15 – 14 – 5 – 6 – 2 – 7 – 8 – 12 – 10 – 11 – 29 – 26
11	2 – 8 – 7 – 1 – 0 – 3 – 5 – 14 – 15 – 13 – 12 – 10 – 11 – 16 – 17
12	7 – 2 – 6 – 5 – 14 – 15 – 21 – 28 – 29 – 11 – 9 – 8 – 12 – 13
13	19 – 21 – 20 – 4 – 22 – 25 – 21 – 28 – 29 – 11 – 9 – 8 – 7 – 1 – 0
14	12 – 8 – 7 – 1 – 0 – 2 – 6 – 15 – 18 – 21 – 28 – 23 – 26 – 29 – 11
15	12 – 8 – 9 – 11 – 29 – 28 – 21 – 25 – 22 – 4 – 3 – 5 – 14 – 15 – 13

Table 3: Best Route Sets (for Operator) of Nigeria Network

Routes	Sequence of Routes
1	21 – 3 – 5 – 14
2	13 – 12 – 10
3	5 – 3
4	18 – 15 – 14
5	10 – 12 – 8
6	2 – 0 – 1
7	20 – 4 – 3 – 5
8	16 – 17 – 12 – 6 – 5
9	9 – 11 – 16
10	29 – 28 – 23 – 26 – 27 – 24
11	2 – 8 – 7

12	24 – 21 – 25 – 22
13	11 – 10
14	19 – 15 – 13
15	2 – 8

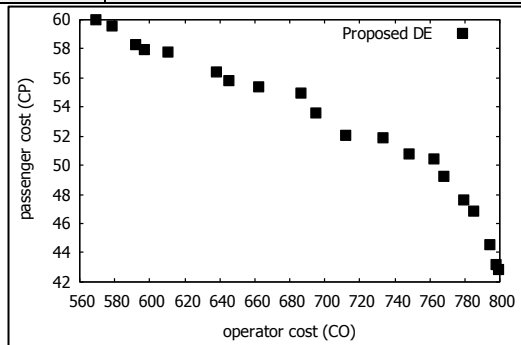


Figure 3: Approximate Pareto Fronts of Nigeria Network for Multiobjective UTRP

4. Discussion

During the implementation on the real network, we note that the reference solution (i.e. the solution operated by the public transportation system of the city) is not available for comparison. In addition, neither optimum solution is known, nor any previous published result exists. Hence, a lower bound on the passenger cost and the operator cost are computed as proposed in [11]. The lower bound (shown in brackets in Table 1) on the passenger cost is based on an ideal situation for passengers travelling on the transit network namely, every passenger can travel to their destinations by the fastest (or shortest) path without any transfers. We calculated the ideal travelling path between each pair of nodes using Dijkstra's algorithm, given the number of nodes, travel time and travel demand between each pair are known. For the operator cost, the lower bound is found by using minimal spanning tree.

In Table 1, from the passenger perspective, all demand are satisfied with at most three transfers, while 2.31% of passenger required more than five transfers from the operator perspective. It can be observed that if the percentage of demand with more than two transfers is considered unsatisfactory (a common practice in literature), then on the operator, it is 33.44% (i.e. $17.67 + 10.27 + 3.19 + 2.31$), which is a very high percentage. Hence, it is reasonable to consider the percentage demand satisfied at more than two transfers (an acceptable situation in real world practice). In addition, the results produced by the proposed algorithm for the passenger ($C_p = 42.88$) is relatively close to the lower bound ($=38.90$). However, from the operator perspective, the operator cost is higher by 22.37%. Furthermore, there is a significant reduction in operator cost from 3891 to 569 when the operator is given the priority. It can be concluded that the proposed DE algorithm is capable of constructing efficient transit routes.

Acknowledgement

This research is supported by Fundamental Research Grant Scheme (FRGS) 01-01-16-1867FR (Ministry of Higher Education, Malaysia).

References

[1] Arbex, R.O. and C.B. da Cunha, 2015. Efficient transit network design and frequencies setting multi-objective optimization by alternating objective genetic algorithm. *Transportation Research Part B: Methodological*, 81: 355–376.

The performance or effectiveness of the proposed algorithm is evaluated using the same parameters adopted by several researchers in the literature (see, [20], [2], [8], [11], [24], and [16]):

- d_0 – the percentage of demand satisfied without any transfers,
- d_j – the percentage of demand satisfied with $j = 1, 2, 3, 4, 5$ transfer(s),
- d_{un} – the percentage of demand unsatisfied ($j \geq 6$),
- C_p – average travel time in minutes per transit user (mpu),
- C_o – total route set length.

[2] Baaj, M.H. and H.S. Mahmassani, 1991. An AI-based approach for transit route system planning and design. *Journal of Advanced Transportation*, 25(2): 187–209.

[3] Bagherian, M., S. Massah, and S. Kermanshahi, 2013. A swarm based method for solving transit network design problem. *Proceedings of the 36th Australasian Transport Research Forum*, Brisbane, Australia.

[4] Bagloe, S.A. and A. Ceder, 2011. Transit network design for actual size networks. *Transportation Research Part B*, 45: 1787–1804.

[5] Beasley, D., D.R. Bull, and R.R. Martin, 1993. An overview of genetic algorithms: part 2, research topics. *University Computing*, 15(4): 170–181.

[6] Buba, A.T. and L.S. Lee, 2016a. Urban transit network design problems: a review of population-based metaheuristics. *Pertanika Journal of Scholarly Research Review*, 2(3): 86–99.

[7] Buba, A.T. and L.S. Lee, 2016b. Differential evolution for urban transit routing problem. *Journal of Computer and Communications*, 4: 11–25.

[8] Chakroborty, P. and T. Wivedi, 2002. Optimal route network design for transit systems using genetic algorithms. *Engineering Optimization*, 34(1): 83–100.

[9] Chew, J.S.C., L.S. Lee, and H.V. Seow, 2013. Genetic algorithm for biobjective urban transit routing problem. *Journal of Applied Mathematics*, 2013. Article ID 698645, 15 pages.

[10] Dijkstra, E.W., 1959. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1(1): 269–271.

[11] Fan, L. and C.L. Mumford, 2010. A metaheuristic approach to the urban transit routing problem. *Journal of Heuristics*, 16(3): 353–372.

[12] Farahani, R.Z., E. Miandoabchi, W.Y. Szeto, and H. Rashidi, 2013. A review of urban transportation network design problems. *European Journal of Operational Research*, 229(2): 281–302.

[13] Guihaire, V. and J-K. Hao, 2008. Transit network design and scheduling: a global review. *Transportation Research Part A: Policy and Practice*, 42(10): 1251–1273.

[14] Ibarra-Rojas, O., F. Delgado, R. Giesen, and J. Muñoz, 2015. Planning, operation, and control of bus transport systems: a literature review. *Transportation Research Part B: Methodological*, 77: 38–75.

[15] John, M.P., C.L. Mumford, and R. Lewis, 2014. An improved multi-objective algorithm for the urban transit routing problem. In: Blum, C., Ochoa, G. (eds) *Evolutionary Computation in Combinatorial Optimisation. EVOCOP Lecture Notes in Computer Science*, vol 8600. Springer, Berlin, Heidelberg.

[16] Kechagiopoulos, P.N. and G.N. Beligiannis, 2014. Solving the urban transit routing problem using a particle swarm optimization based algorithm. *Applied Soft Computing*, 21: 654–676.

[17] Kepaptsoglou, K. and M. Karlaftis, 2009. Transit route network design problem: review. *Journal of Transportation Engineering*, 135: 491–505.

- [18] Kiliç, F. and M. Gök, 2014. A demand based route generation algorithm for public transit network design. *Computers and Operations Research*, 51: 21–29.
- [19] Lee, Y.J. and V.R. Vuchic, 2005. Transport network design with variable demand. *Journal of Transportation Engineering*, 131(1): 1–10.
- [20] Mandl, C.E., 1980. Evaluation and optimization of urban public transportation networks. *European Journal of Operational Research*, 5(6): 396–404.
- [21] Mumford, C.L., 2013. New heuristic and evolutionary operators for the multi-objective urban transit routing problem. 2013 IEEE Congress on Evolutionary Computation, 939–946.
- [22] Nayeem, M.A., M.K. Rahman, and M.S. Rahman, 2014. Transit network design by genetic algorithm with elitism. *Transportation Research Part C: Emerging Technologies*, 46: 30–45.
- [23] Ngamchai, S. and D.J. Lovell, 2003. Optimal time transfer in bus transit route network design using a genetic algorithm. *Journal of Transportation Engineering*, 129(5): 510–521.
- [24] Nikolić, M. and D. Teodorović, 2013. Transit network design by bee colony optimization. *Expert Systems with Applications*, 40(15): 5945–5955.
- [25] Pattnaik, S., S. Mohan, and V. Tom, 1998. Urban bus transit route network design using genetic algorithm. *Journal of Transportation Engineering*, 124(4): 368–375.
- [26] Storn, R. and K. Price, 1995. *Differential Evolution – A Simple and Efficient Adaptive Scheme for Global Optimization over Continuous Spaces*, International Computer Science Institute, Berkeley. Berkeley, CA.
- [27] Zhao, H. and R. Jiang, 2015. The memetic algorithm for the optimization of urban transit network. *Expert Systems with Applications*, 42(7): 3760–3773.