Journal of Computer and Communications, 2016, 4, 11-25
http://www.scirp.org/journal/jcc
ISSN Online: 2327-5227
ISSN Print: 2327-5219

Differential Evolution for Urban Transit Routing Problem

Ahmed Tarajo Buba[1], **Lai Soon Lee**[1,2*]

[1]Department of Mathematics, Faculty of Science, Universiti Putra Malaysia, Selangor, Malaysia
[2]Laboratory of Computational Statistics and Operations Research, Institute for Mathematical Research, Universiti Putra Malaysia, Selangor, Malaysia
Email: *lls@upm.edu.my

How to cite this paper: Buba, A.T. and Lee, L.S. (2016) Differential Evolution for Urban Transit Routing Problem. *Journal of Computer and Communications*, **4**, 11-25.
http://dx.doi.org/10.4236/jcc.2016.414002

Received: September 5, 2016
Accepted: November 8, 2016
Published: November 11, 2016

Abstract

The urban transit routing problem (UTRP) involves the construction of route sets on existing road networks to cater for the transit demand efficiently. This is an NP-hard problem, where the generation of candidate route sets can lead to a number of potential routes being discarded on the grounds of infeasibility. This paper presents a new repair mechanism to complement the existing terminal repair and the make-small-change operators in dealing with the infeasibility of the candidate route set. When solving the UTRP, the general aim is to determine a set of transit route networks that achieves a minimum total cost for both the passenger and the operator. With this in mind, we propose a differential evolution (DE) algorithm for solving the UTRP with a specific objective of minimizing the average travel time of all served passengers. Computational experiments are performed on the basis of benchmark Mandl's Swiss network. Computational results from the proposed repair mechanism are comparable with the existing repair mechanisms. Furthermore, the combined repair mechanisms of all three operators produced very promising results. In addition, the proposed DE algorithm outperformed most of the published results in the literature.

Keywords

Urban Transit, Routing, Network, Repair Mechanism, Differential Evolution

1. Introduction

In recent years, the rapid growth of population and urbanization of large cities with its attendant consequences such as increase in travel demand, traffic congestion, energy consumption, noise level, and air pollution have been a major source of concern to the urban transportation planners and practitioners, particularly in developing and emerging countries. A reliable option to handle some of these problems is to make use of the public transit systems in the urban areas. The public transit systems which include buses, trams, trains and underground or metro services aim to satisfy the travel demand of urban commuters safely, affordably, and efficiently. Although achieving a highly attractive public transit usage is a very complex issue, however, designing an operationally and economically efficient public transit network can be a significant attribute. Furthermore, the urban area's social, economic, and physical structure can be enhanced through the construction of efficient public transit networks.

In a general sense, designing bus network is commonly characterized by mutually conflicting factors across different planning horizons. For instance, higher level of service in terms of shorter waiting time, faster travel time, and minimum number of transfers is provided to the transit users in the presence of more routes and buses needed. In other words, the network topology has a direct impact on the costs of the users, benefits of the transit operators, and the relevant social costs. In bus transit system, the literature identifies five stages for designing a public transportation system [1]: route network design, frequency setting, timetable design, fleet assignment and crew assignment. Most previous approaches have attempted to handle these stages sequentially in the real systems, because each stage is NP-hard in its own and this creates several sources of complexity. At the same time, these decisions are made for different planning horizons, whether the context of the planning is strategic, tactical or operational.

The urban transit routing problem (UTRP) which corresponds to the first stage of the bus planning process involves the construction of transit routes on an existing road network based on the travel demand and corresponding link travel times subject to given constraints and requirements such that the routes optimize the desired objective(s) defined by the stakeholders (including users, operators, and society). It represents the single very strategic planning step in the urban bus planning process that seeks to balance the competing objectives of minimizing both passenger and operator costs ([1] [2]).

The contributions of this study are in two folds. First, a new repair mechanism called sub-route reversal repair mechanism is designed to complement the existing terminal repair [3] and the make-small-change operator [4] in dealing with the infeasibility of the route sets. The proposed repair mechanism managed to repair some of the candidate route sets which are deemed as infeasible by both repair mechanisms. In the second contribution, a new algorithm based on differential evolution (DE) has been developed to solve the UTRP. To the best of our knowledge, there is no study of DE on UTRP. The decision to develop the DE is due to its flexibility, simplicity, robustness and its wide applications to many other constrained optimization problems with promising results ([5] [6]). For a fair comparison with other approaches in the litarature, the same performance evaluation parameters adopted in the literature with known benchmark problems are used.

The rest of the paper is organized as follows: Section 2 presents a brief literature review of the solution approaches for UTRP. The mathematical formulation of UTRP is given in Section 3. The details of the proposed repair mechanism and the DE algorithm are described in Section 4 and 5 respectively. Computational results and discussions are presented in Section 6. Finally, conclusions and the directions for future research are provided in Section 7.

2. Literature Review

A search in the literature provides various heuristic and metaheuristic algorithms for the optimization of UTRP. A heuristic algorithm to find the optimal route sets is first proposed by [7]. The methodology consists of two major stages including the generation of feasible candidate route sets, and the selection of the optimal route set. The Swiss network used in [7] has since become the only publicly available benchmark dataset for UTRP.

Reference [1] introduced a model for tackling the routing and scheduling together. The methodology consists of two-level routines for constructing the initial candidate route set and testing the candidate route set. The user's viewpoint is considered at the first level, while both users and operators point of view is considered at the second level. A hybrid solution approach for the network design problem by incorporating the knowledge of experts and optimization techniques is presented in [8]. The AI-based approach is composed of three main components including route generation algorithm, analysis procedure, and route improvement algorithm. Experiments were conducted to evaluate the performance of the route generation algorithm.

Over the past decade, genetic algorithm (GA) has gain significant success in determining near optimal solutions for the UTRP ([2] [9]-[12]). A GA is proposed by [2] in designing efficient and optimal sets of routes from initial candidate route sets based on a given road network and data on travel demand. A GA to tackle the UTRP giving priority to the passenger cost is presented in [9]. The computational results on the benchmark Mandl's Swiss network outperforms most approaches in the literature. Reference [10] developed two models of GA to address the problem. The aim is to optimize the size of passengers satisfied, the overall travel time of served passengers, and the total number of transfers. The GA with elitism produced some competitive results, while the GA with increasing population outperforms all previous results in the literature. However, their results cannot be used as a direct comparison with others as the constraints on the maximum number of nodes (*i.e.* 8 nodes) in a route is relaxed as compared to other approaches in the literature.

A hill climbing heuristic and simulated annealing (SA) is developed in [4] for solving the UTRP. Three issues including representation, initialization procedures, and neighborhood moves that need to be considered to ensure the success of metaheuristics are highlighted. A bee colony optimization (BCO) algorithm for solving the UTRP is proposed in [13]. Their approach consist of generating initial candidate solutions using a greedy heuristic, then an improvement version of the BCO is introduced to construct optimal route sets. Experiments are conducted on benchmark Mandl's Swiss network confirming that the BCO approach can produced high quality solutions. The particle swarm optimization (PSO) algorithm developed in [14] and [15] for solving the problem gives proper emphasis on both candidate solution representations and evaluation approach. The aim is optimize the coverage index subject to operator cost at the upper level. The methodology consists of two main elements: a candidate route construction module, and a discrete PSO algorithm to determine the optimal route set.

A number of studies employed multiobjective models to solve the UTRP and the corresponding vehicle frequencies setting simultaneously ([16]-[23]). Reference [16] proposed a GA for solving the problem through fixed and variable string length coding. Based on some descriptors considered in the study, the fixed string length coding produced better results compared to the variable length coding. The GA proposed by [17] optimized the location of bus route and its corresponding headway. The proposed GA is found to converge efficientlyto optimum as confirmed by the solution from exhaustive search algorithm. A new approach to evaluate the fitness function values in GA for optimization of bus network is given in [18]. The aim is to develop a heuristic that determines the best bus route network of which both demand and transport offer are satisfied. An experiment is conducted and the model is tested on a realistic network. The GA developed by [19] tackled the bi-objective UTRP in optimizing the user and operator costs. The proposed GA introduces an adding-node mechanism to correct an infeasible solution. The genetic operators include effective route crossover and identical-point mutation. The computational results of the proposed GA tested on benchmark Mandl's Swiss network outperformed the previous best pulished results in most cases.

Reference [20] proposed a SA to solve the UTRP by considering the level of distribution node. The methodology comprised of initialization, transit trips assignment, and a SA algorithm that selects the optimum solution. To evaluate the quality of the solution obtained by the proposed SA, a GA solution is utilized as the benchmark. A tabu search (TS) is developed in [21] to optimize the problem. The methodology consists of three major components: initialization, demand assignment, and TS algorithm that determine the optimum. Computational results obtained outperform the GA which is used as the benchmark as in [20]. The TS employed in [22] analyzed an urban transportation problem in northern Spain. The aim is to optimize the level of service. The methodology consists of line design and assignment of buses such that a local search and TS strategies are modified alternatively at the two decision levels. The result obtained is far better than the existing tools used by transport authorities.

3. Mathematical Formulation

In this study, the UTRP is solved from the passengers' point of view. Generally, passengers would prefer to travel to their destination within the shortest journey time possible, but avoiding the discomfort associated with too many transfers. The passengers' cost for a route set, R is defined as the average journey time over all passengers. It is a difficult task to optimize the transit network where the journey time includes vehicle travel time, waiting time, transfer time, and transfer penalties. Based on the studies by [3] and [19], minimizing the passengers' cost is often considered as the most significant objective in bus network design with inelastic demand.

In this study, the UTRP is defined as follows: given a road network represented by the graph, $G = (N, A)$, where N is the set of nodes representing the demand points (bus stops) and A is the set of edges (links) representing the street segments. Let λ_{ij} denote the shortest journey time of any pair of nodes (i, j) of route set R, calculated based on the Dijkstra's algorithm [24], n is the number of nodes of the route set R, d_{ij} represents the travel demand between nodes i and j. The proxy for passenger cost for a route set R is the mean journey time over all passengers [3]. The objective of the UTRP is to find a set of route network that achieves the minimum cost Equation (1) while meeting all the requirements and constraints of (1)-(8):

$$\min z = \frac{\sum_{i,j=1}^{n} d_{ij}\lambda_{ij}}{\sum_{i,j=1}^{n} d_{ij}} \tag{1}$$

subject to the following constraints:

1) Each route should have a minimum and a maximum length (a minimum length to ensure connectivity of the route network while a maximum length to aid bus schedule adherence).

2) There must be exactly number of routes in a route set R predefined by the transit operator due to resource limitation.

3) The transit graph must be connected, so that there should be a path connecting any two nodes where any passenger using the route network can travel between any two nodes.

4) Each route in the route set is free from repeated nodes. Hence, no cycles or backtracks should be allowed in the individual routes.

5) All nodes must be included in the route set to form a complete route set.

6) The demand, travel time, and distance matrices are symmetrical along the same route.

7) The demand level is inelastic throughout the period of the study and passenger choice of routes is based on the shortest travel time.

8) The policy headway is not considered, it is assumed there are adequate vehicles and capacity.

4. Sub-Route Reversal Repair Mechanism

Based on the literature, heuristics and shortest path algorithms have been commonly utilized to generate the candidate route sets during the initialization procedure. According to [4], it has not been justified which of the two approaches will yield the best initial solution. The quality of the initial solution is assessed in terms of coverage, connectivity, and diversity of the candidate route sets. The use of heuristics and shortest path algorithms often resulted in many initial candidate route sets being rejected due to infeasibility. A feasible candidate route set is characterized by connected route sets and every node presents in the original transit network is included. In most cases, a repair mechanism is employed to reduce the infeasibility of the candidate route sets. The repair mechanisms currently available in the literature are the terminal repair [3] and make-small-change [4]. However, there are still far from being efficient. With the aim to improve the quality of the initial candidate route sets in terms of the feasibility, a new repair mechanism, called sub-route reversal repair mechanism is proposed to complement the existing repair mechanisms mentioned above. The steps of the proposed repair mechanism in dealing with an infeasible route set are as follows:

STEP 1: Compile a list of the missing nodes, $\check{N} = \{\check{n}_1, \check{n}_2, \dots, \check{n}_m\}$ from the constructed infeasible route set, $\check{R} = \{\check{r}_1, \check{r}_2, \dots, \check{r}_s\}$.

STEP 2: The first route from $\check{R}$ is selected as the current route, $\check{r}_i$ ($i = 1$).

STEP 3: The first node from $\check{N}$ is selected as the current missing node, $\check{n}_j$ ($j = 1$).

STEP 4: **IF**, $\check{r}_i$ contains a neighbor (*i.e.* two nodes are directly connected) of $\check{n}_j$, go to **STEP 5**, **ELSE**, go to **STEP 6**.

STEP 5: Identify the neighbor x, of $\check{n}_j$ in $\check{r}_i$. The route $\check{r}_i$ is divided into two sub-routes, s_1 and s_2 for two cases:

Case 1: the neighbor x is located at the end of s_1. Reverse the sequence of nodes in s_1 so the x becomes the first node. The new route is built from the reversed s_1 by appending the nodes from s_2 (if feasible) continuously until the neighborhood relation is exhausted or the maximum number (*i.e.* 8) of nodes for a route is satisfied. This new route is labeled as $\check{r}_i^1$.

Case 2: the neighbor x is located at the beginning of s_2. Reverse the sequence of nodes in s_2 so the x is located at the end of the node sequence in this reversed s_2 sub-route. The new route is built from the reversed s_2 by appending the nodes from s_1 (if feasible) continuously until the neighborhood relation is exhausted or the maximum number (*i.e.* 8) of nodes for a route is satisfied. This new route is labeled as $\check{r}_i^2$.

The new $\check{r}_i^* = \max\{\check{r}_i^1, \check{r}_i^2\}$. Breaking ties arbitrarily. Update $\check{N}$ by removing the $\check{n}_j$ from the list. Update $\check{r}_i$ in $\check{R}$ with the new route $\check{r}_i^*$. Go to **STEP 2**.

STEP 6: **IF**, the current route, $\check{r}_i$ is not the last route of $\check{R}$, set $i = i + 1$, go to **STEP 4**. **ELSE**, go to **STEP 7**.

STEP 7: The missing node, $\check{n}_j$ could not be inserted in $\check{R}$. Hence, the $\check{R}$ is infeasible. **STOP**.

The steps are repeated until all missing nodes are inserted into the route set (*i.e.* the infeasibility is repaired), or when the STEP 7 is invoked (*i.e.* infeasible).

To further enhance the efficiency in dealing with the infeasibility of the route sets, a combined repair mechanism is used. The combined repair mechanism work in such a way that the three repair mechanisms, terminal repair, make-small-change and sub-route reversal are performed in a sequential order. The terminal repair will be applied first to repair the infeasibility. If the operator failed to repair the infeasible route set, the make-small-change will be applied next, follow by the sub-route reversal operator. The efficiency of the proposed repair mechanism is investigated in Section 6.2.

5. Differential Evolution

In this section, the classic DE and the proposed DE framework for solving the UTRP are discussed. The DE is a variant of GA that is originally introduced by [25] for global optimization problems over continuous space. The DE algorithm utilizes three major operators similar to GAs including the mutation, crossover and selection operators. However, DE depends heavily on mutation as a primary search mechanism to distinguish it from the GA. The mutation operator of DE involves the computation of weighted differences of randomly sampled pairs of solutions in the population such that the exploration of the search space is favored at the initial stage of the evolution process. Subsequently, exploitation of the search space is favored as evolution advances as a result of mutation operator. Therefore, during the implementation, the DE adapts the search step automatically through the mutation to achieve the best value. The DE algorithm utilizes a uniform crossover that can select child vector from one parent more often than the other to construct trial vectors. The crossover operator efficiently exchanges information between successful combinations, whereby the most promising area of the search space is located for an optimum.

In the DE algorithm, a mutant vector is created for each target vector in the population. Then the crossover operator is carried out between the mutated population and the target population in order to give rise to a population of trial vectors. Next, the selection operator is introduced for the comparison of the target and trial population based on the fitness value. Finally, better vectors constitute the members of the population for the next generation. Through repeated cycles of the evolutionary operators the DE directs the population towards the neighborhood of the global optimum.

Proposed Differential Evolution Algorithm Framework

While employing the DE algorithm framework, it is likely that infeasible vectors will be generated due to the evolutionary operators (mutation and crossover). Some studies in the literature considered such vectors to be rejected and the whole initialization procedure together with feasibility checks is repeated to construct a feasible solution to replace it ([3] [19] and [26]). In this paper, we employed the combined repair mechanisms as described in Section 4 to deal with the infeasibility. The steps of the proposed DE framework for solving the UTRP are structured as follows:

STEP 1:	The DE algorithm starts with a population of N_p solution vectors initially generated based on the construction algorithm [3] by incorporating the combined repair mechanisms to increase the number of feasible vectors.
STEP 2:	During the generation G, for a *Target* vector $X_{i,G} = (x_{1,i,G}, x_{2,i,G}, \ldots, x_{d,i,G})$, where $i = 1$, d represents d-components in the d-dimensional space; a random vector is selected from the population (except the selected *Target* vector) and an identical point mutation proposed in [19] is applied on the random vector to generate a *Noisy Random* vector, $V_{i,G}$. In the case where the resulted Noisy Random vector, $V_{i,G}$ is infeasible, the combined repair mechanism is introduced to correct the infeasibility.

Continued

STEP 3:	To increase the diversity of the *Target* and *Noisy Random* vectors, the uniform route crossover [27] is introduced. In this work, a pair of *Trial* vectors, $U_{i,G}$ is generated through selecting the vector component values either from the *Target* vector, $X_{i,G}$ or the *Noisy Random* vector, $V_{i,G}$ using a 0/1 crossover mask where each sub-route in the *Trial* vector is constructed by copying the corresponding sub-route either from the *Target* or the *Noisy Random* vector. In the case where the resulted *Trial* vector(s), $U_{i,G}$ is/are infeasible, the combined repair mechanism is introduced.
STEP 4:	After the crossover, the objective function values corresponding to the *Trial* vectors, $U_{i,G}$ are evaluated and compared with that of the *Target* vector, $X_{i,G}$.
STEP 5:	An elitism selection strategy is employed, where the best vector with the lowest fitness value between the *Target* vector, $X_{i,G}$ and the *Trial* vectors, $U_{i,G}$ will be selected for the next generation. Repeat **STEP 2** - **STEP 4** for $i = 2, \ldots N_p$ to complete one generation.

STEP 2 - STEP 5 are repeated until the termination criterion (e.g., maximum generation, execution time, etc.) is met. The framework of the proposed DE for solving the UTRP is shown in **Algorithm 1**.

6. Computational Experiments and Discussion

6.1. Benchmark Data and Experimental Design

The proposed DE algorithm is utilized to design the best transit route network of Mandl's Swiss network [7] (see **Figure 1**) which is the benchmark network for demonstrating the effectiveness and efficiency of an algorithm. This network consists of 15 demand points (nodes) within a 33 minutes shortest travel time between the two farthest nodes and 21 links, with a total travel demand of 15570 units daily. Many variations

Algorithm 1. DE for UTRP.

```
Generate N_p candidate route set based on heuristic in [3] with combined repair mechanism
for i: = 1 to N_p
    fitness evaluation (Equation (1))
end for
for n: = 1 to G
    for i: = 1 to N_p
        set Target vector = X_{i,n}
        select randomly a vector (except the selected Target vector, X_{i,n}) in the population
        apply identical point mutation to generate a Noisy Random vector, V_{i,n} (repair if infeasible)
        apply uniform crossover between X_{i,n} and V_{i,n} to generate a pair Trial vectors, U_{i,n} (repair if
        infeasible)
        fitness evaluation of U_{i,n} (Equation (1))
        elitism selection
        if Trial vector fitness ≤ Target vector fitness
            new_population [i] = Trial vector, U_{i,n}
        else
            new_population [i] = Target vector, X_{i,n}
    end for
    N_p = new_population
end for
return BEST
```

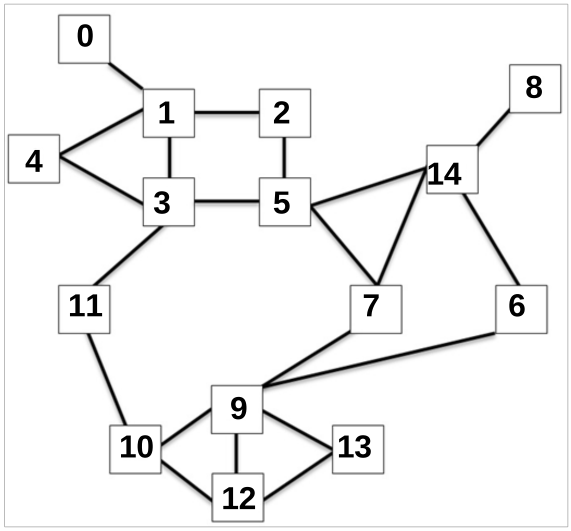

Figure 1. Mandl's Swiss road network.

of the UTRP (*i.e.* different objectives) are solved by [2] [4] [9] [10] [13] [14], and [28]. The performance parameters together with their solutions will be used as the benchmark results to validate the proposed DE algorithm.

The proposed algorithm is coded in Python programming language and executed on a 1.60 GHz Intel Core™ i 5 - 4200 CPU with 4.00 GB of RAM. The performance of the proposed DE algorithm is evaluated using the following parameters adopted by many researchers in the literature ([4] [7] [9] [13] [14] and [28]):

- do—the percentage of demand satisfied without any transfers,
- d1—the percentage of demand satisfied with one transfer,
- d2—the percentage of demand satisfied with two transfers,
- dun—the percentage of demand unsatisfied,
- ATT—average travel time in minutes per transit user.

The travel time includes vehicle travel time, waiting time, transfer time, and transfer penalties. Based on the literature, the transfer penalty is fixed at 5 minutes per passenger. The minimum and maximum of nodes in a route is also fixed at 2 and 8 nodes respectively. We performed comparison in four scenarios: 4, 6, 7 and 8 routes in each route set. The best values of the parameters are computed.

6.2. Computational Experiments of Repair Mechanisms

To investigate the efficiency of the proposed sub-route reversal repair mechanism in dealing with the infeasible candidate route set, we compared it with the terminal repair [3], make-small-change [4], and the combined repair mechanism. We performed the experiments on benchmark Mandl's Swiss network by generating a population of 500 infeasible candidate route sets using the heuristic proposed in [3] in all four cases with route set sizes of 4, 6, 7, and 8 routes respectively. For each case, 10 runs are recorded to obtain the descriptors for each repair mechanism as shown in Table 1. The descriptors represent the average, minimum, and maximum number of infeasible candidate route

Table 1. The comparison of infeasible route sets repaired by the repair mechanisms.

Case	Number of Routes	Repair Mechanism	Average	Minimum	Maximum	CPU Time (sec)
I	4	Terminal Repair	47	34	68	0.013
		Make-Small-Change	53	39	69	0.052
		Sub-Route Reversal	56	47	68	0.040
		Combined Repair	**139**	**126**	**157**	**0.001**
II	6	Terminal Repair	149	133	170	0.029
		Make-Small-Change	206	193	220	0.490
		Sub-Route Reversal	169	157	181	0.032
		Combined Repair	**292**	**277**	**306**	**0.002**
III	7	Terminal Repair	201	184	219	0.029
		Make-Small-Change	293	267	317	0.066
		Sub-Route Reversal	223	210	234	0.025
		Combined Repair	**352**	**335**	**377**	**0.001**
IV	8	Terminal Repair	231	207	250	0.020
		Make-Small-Change	356	319	376	0.042
		Sub-Route Reversal	265	242	284	0.025
		Combined Repair	**379**	**347**	**399**	**0.012**

sets that were repaired by the corresponding repair mechanism. It can be seen in Table 1 that the results from the proposed sub-route reversal repair mechanism is performed better than the terminal repair in all cases but only slightly better than the make-small-change in one of the cases. Although the make-small-change repaired more infeasible route sets than the proposed repair mechanism, but its average CPU time is the highest in all cases.

The results obtained through the combined repair mechanism for all cases are very promising. The mechanism significantly outperformed all other repair mechanisms in terms of the number of infeasible route sets repaired and with the fastest CPU time. Therefore, it is only logical to implement the combined repair mechanism in both the heuristic [3] and the proposed DE algorithm when dealing with the infeasible route sets.

6.3. Comparative Results of Differential Evolution

In this section, the UTRP is solved by the proposed DE algorithm from the viewpoint of the passengers' cost. We conducted the computational experiments for each of the four cases to obtain the best route set, evaluated based on the performance parameters explained in Section 6.1. A population of 20 vectors and 200 generations is used for the computational experiments based on the initial experiments conducted during the development of the proposed algorithm. For each case, the proposed DE algorithm is performed for 10 runs and the best result is reported in Table 2, column 12. The previous best solutions found in the literature are reported in column 3 to column 11. The best route sets for all cases found by the proposed DE algorithm are also presented in Table 3.

It is important to note that, some of the approaches in the literature utilize the same formulation for the modeling of the UTRP and attempt to optimize the equivalent objective function. For instance, [10] proposed two algorithms (GAWE and GAWIP) to handle the UTRP in a similar manner as in [13] but by relaxing the constraint on route length (max. 8 nodes) in their approach to achieve the best results so far (see Table 2, column 10 and 11). However, [14] confirmed computationally that if the number of nodes per route is set to a value greater than 8, the percentage transfer demands satisfied directly (do) is increased. On the other hand, as the number of nodes per route is increased, the number of transfer demands satisfied with one transfer or with two transfers is decreased. However, in practice, increasing the length of route alone is not enough to improve the quality of transport services.

Table 2. Comparison results (best route sets) of Mandl's Swiss network.

Number of routes	Parameters	1	2	3	4	5	6	7	**8***	**9***	**10**
4	do	69.94	n/a	86.86	93.26	92.42	92.10	91.84	96.14	95.83	**94.28**
	d1	29.93	n/a	12.00	6.74	6.81	7.19	7.64	3.47	3.60	**5.72**
	d2	0.13	n/a	1.14	0.00	0.77	0.71	0.52	0.39	0.57	**0.00**
	dun	0.00	n/a	0.00	0.00	0.00	0.00	0.00	0.00	0.00	**0.00**
	ATT	12.90	n/a	11.90	11.37	10.51	10.51	10.64	10.49	10.35	**10.36**
6	do	n/a	78.61	86.04	91.52	95.57	95.63	96.21	98.39	98.91	**98.01**
	d1	n/a	21.39	13.96	8.48	4.43	4.37	3.47	1.61	1.09	**1.99**
	d2	n/a	0.00	0.00	0.00	0.00	0.00	0.32	0.00	0.00	**0.00**
	dun	n/a	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	**0.00**
	ATT	n/a	11.86	10.30	10.48	10.28	10.23	10.23	10.14	10.10	**10.01**
7	do	n/a	80.99	89.15	93.32	96.60	98.52	97.17	99.17	99.55	**98.07**
	d1	n/a	19.01	10.85	6.36	3.40	1.48	2.83	0.83	0.45	**1.93**
	d2	n/a	0.00	0.00	0.32	0.00	0.00	0.00	0.00	0.00	**0.00**
	dun	n/a	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	**0.00**
	ATT	n/a	12.50	10.15	10.42	10.25	10.15	10.16	10.07	10.07	**10.21**
8	do	n/a	79.97	90.38	94.54	97.50	98.97	97.75	99.87	99.87	**98.59**
	d1	n/a	20.03	9.62	5.46	2.50	1.03	2.25	0.13	0.13	**1.41**
	d2	n/a	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	**0.00**
	dun	n/a	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	**0.00**
	ATT	n/a	11.86	10.46	10.36	10.17	10.09	10.13	10.03	10.04	**10.01**

Note: n/a = not available, *constraint of maximum 8 nodes per route is relaxed, 1: Heuristic by [7], 2: AI by [28], 3: GA by [2], 4: Metaheuristic by [4], 5: GA by [9], 6: BCO by [13], 7: PSO by [14], 8: GA with Elitism by [10], 9: GA with Increasing Population by [10], 10: Proposed DE algorithm.

Table 3. Best route sets generated from Mandl's Swiss network by the proposed DE algorithm.

Case	Number of Routes	Description of routes
I	4	13-9-10-11-3-1-2-5
		0-1-2-5-7-14-6-9
		8-14-5-7-9-13-12-10
		0-1-4-3-11-10-9-12
II	6	13-12-10-11-3-4-1-0
		9-6-14-7-5-2-1-0
		12-13-9-10-11-3-5-7
		4-3-1-2-5-14-7-9
		8-14-5-7-9-12-10-11
		6-9-10-11-3-5-2
III	7	0-1-3-5-7-9-6
		8-14-6-9-10-11-3-4
		0-1-2-5-3-11-10-12
		0-1-4-3-11-10-9-7
		2-1-3-5-7-14-6-9
		9-10-11-3-5-2-1-4
		13-12-10-9-6-14-8
IV	8	6-14-7-5-3-4-1-2
		1-4-3-11-10-12-13
		0-1-3-11-10-9-6
		8-14-7-5-3-11-10-9
		11-10-12-13-9-6-14-8
		8-14-5-2-1-4-3
		9-10-11-3-4-1-2-5
		12-13-9-7-5-2-1-0

The quality of a route set is assessed by considering the parameters mentioned in Section 6.1. In **Table 2**, the proposed DE algorithm performed the best for 4 and 6 routes while in 7 and 8 routes, the DE algorithm is ranked second after [13]. However, the best ATT recorded by the DE algorithm is better than [13] in the case of 8 routes. Note that the results achieved by [10] should not be used as a direct comparison since the constraints on the maximum number of nodes (*i.e.* 8 nodes) in a route is relaxed as compared to other approaches in **Table 2**.

In all four cases considered in this study, the DE produced results with maximum only one transfer required to satisfy all the travel demand. In addition, as efficient route set is defined by the values of dun (0.00), do (as high as possible) and ATT (as low as possible), it can be concluded that the route sets constructed by the proposed DE algorithm for all cases are highly efficient as compared to other approaches in the literature.

7. Conclusion

In this paper, a sub-route repair mechanism that complements the terminal repair and make-small-change repair mechanisms in dealing with the infeasible route sets is proposed. To further enhance the efficiency, a combined repair mechanism is developed where all three repair mechanisms: terminal repair, make-small-change and sub-route reversal, are implemented in a sequential order. The combined repair mechanism significantly outperformed all other repair mechanisms with the fastest CPU time. In addition, a new DE algorithm that constructs efficient route networks for UTRP has been developed. Computational experiments performed on the benchmark Mandl's Swiss network show that the proposed DE algorithm is competitive to the other approaches published in the literature. In real sense, the efficiency of the urban transport system depends not solely on the topology of the transit route network, but also on the operating frequency of the routes. Consequently, the application of the proposed model to handle the UTRP and the operating frequency simultaneously, as well as improving the effectiveness of the sub-route reversal repair mechanism will be the focus of the future work.

References

[1] Ceder, A. and Wilson, N.H.M. (1986) Bus Network Design. *Transportation Research Part B: Methodological*, **20**, 331-344. http://dx.doi.org/10.1016/0191-2615(86)90047-0

[2] Chakroborty, P. and Wivedi, T. (2002) Optimal Route Network Design for Transit Systems Using Genetic Algorithms. *Engineering Optimization*, **34**, 83-100. http://dx.doi.org/10.1080/03052150210909

[3] Mumford, C.L. (2013) New Heuristic and Evolutionary Operators for the Multi-Objective Urban Transit Routing Problem. 2013 *IEEE Congress on Evolutionary Computation*, Cancun, Mexico, 20-23 June 2013, 939-946. http://dx.doi.org/10.1109/CEC.2013.6557668

[4] Fan, L. and Mumford, C.L. (2010) A Metaheuristic Approach to the Urban Transit Routing Problem. *Journal of Heuristics*, **16**, 353-372. http://dx.doi.org/10.1007/s10732-008-9089-8

[5] Vasan, A. and Simonovic, S.P. (2010) Optimization of Water Distribution Network Using Differential Evolution. *Journal of Water Resources Planning and Management*, **136**, 279-287. http://dx.doi.org/10.1061/(ASCE)0733-9496(2010)136:2(279)

[6] Das, S. and Suganthan, P.N. (2011) Differential Evolution: A Survey of the State-of-the-Art. *IEEE Transactions on Evolutionary Computation*, **15**, 4-31. http://dx.doi.org/10.1109/TEVC.2010.2059031

[7] Mandl, C.E. (1980) Evaluation and Optimization of Urban Public Transportation Networks. *European Journal of Operational Research*, **5**, 396-404. http://dx.doi.org/10.1016/0377-2217(80)90126-5

[8] Baaj, M.H. and Mahmassani, H.S. (1995) Hybrid Route Generation Heuristic Algorithm for the Design of Transit Networks. *Transportation Research Part C: Emerging Technologies*, **3**, 31-50. http://dx.doi.org/10.1016/0968-090X(94)00011-S

[9] Chew, J.S.C. (2012) Genetic Algorithms for Urban Transit Routing Problems. M.Sc. Thesis, Universiti Putra Malayisia, Malaysia.

[10] Nayeem, M.A., Rahman, M.K. and Rahman, M.S. (2014) Transit Network Design by Genetic Algorithm with Elitism. *Transportation Research Part C: Emerging Technologies*, **46**,

30-45. http://dx.doi.org/10.1016/j.trc.2014.05.002

[11] Ngamchai, S. and Lovell, D.J. (2003) Optimal Time Transfer in Bus Transit Route Network Design Using a Genetic Algorithm. *Journal of Transportation Engineering*, **129**, 510-521. http://dx.doi.org/10.1061/(ASCE)0733-947X(2003)129:5(510)

[12] Tom, V. and Mohan, S. (2003) Transit Route Network Design Using Frequency Coded Genetic Algorithm. *Journal of Transportation Engineering*, **129**, 186-195. http://dx.doi.org/10.1061/(ASCE)0733-947X(2003)129:2(186)

[13] Nikolić, M. and Teodorović, D. (2013) Transit Network Design by Bee Colony Optimization. *Expert Systems with Applications*, **40**, 5945-5955. http://dx.doi.org/10.1016/j.eswa.2013.05.002

[14] Kechagiopoulos, P.N. and Beligiannis, G.N. (2014) Solving the Urban Transit Routing Problem Using a Particle Swarm Optimization Based Algorithm. *Applied Soft Computing*, **21**, 654-676. http://dx.doi.org/10.1016/j.asoc.2014.04.005

[15] Bagherian, M., Massah, S. and Kermanshahi, S. (2013) A Swarm Based Method for Solving Transit Network Design Problem. *Proceedings of the* 36*th Australasian Transport Research Forum* (*ATRF*), Brisbane, Australia, 2-4 October 2013. http://atrf.info/papers/2013/2013_bagherian_massah_kermanshahi.pdf

[16] Pattnaik, S., Mohan, S. and Tom, V. (1998) Urban Bus Transit Route Network Design Using Genetic Algorithm. *Journal of Transportation Engineering*, **124**, 368-375. http://dx.doi.org/10.1061/(ASCE)0733-947X(1998)124:4(368)

[17] Chien, S., Yang, Z. and Hou, E. (2001) Genetic Algorithm Approach for Transit Route Planning and Design. *Journal of Transportation Engineering*, **127**, 200-207. http://dx.doi.org/10.1061/(ASCE)0733-947X(2001)127:3(200)

[18] Bielli, M., Caramia, M. and Carotenuto, P. (2002) Genetic Algorithms in Bus Network Optimization. *Transportation Research Part C*: *Emerging Technologies*, **10**, 19-34. http://dx.doi.org/10.1016/S0968-090X(00)00048-6

[19] Chew, J.S.C., Lee, L.S. and Seow, H.V. (2013) Genetic Algorithm for Biobjective Urban Transit Routing Problem. *Journal of Applied Mathematics*, **2013**, Article ID: 698645. http://dx.doi.org/10.1155/2013/698645

[20] Fan, W. and Machemehl, R.B. (2006) Using a Simulated Annealing Algorithm to Solve the Transit Route Network Design Problem. *Journal of Transportation Engineering*, **132**, 122-132. http://dx.doi.org/10.1061/(ASCE)0733-947X(2006)132:2(122)

[21] Fan, W. and 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. http://dx.doi.org/10.1111/j.1467-8667.2008.00556.x

[22] Pacheco, J., Alvarez, A., Casado, S. and González-Velarde, J.L. (2009) A Tabu Search Approach to an Urban Transport Problem in Northern Spain. *Computers & Operations Research*, **36**, 967-979. http://dx.doi.org/10.1016/j.cor.2007.12.002

[23] Mauttone, A. and Urquhart, M.E. (2009) A Multi-objective Metaheuristic Approach for the Transit Network Design Problem. *Public Transport*, **1**, 253-273. http://dx.doi.org/10.1007/s12469-010-0016-7

[24] Dijkstra, E.W. (1959) A Note on Two Problems in Connexion with Graphs. *Numerische Mathematik*, **1**, 269-271. http://dx.doi.org/10.1007/BF01386390

[25] Storn, R. and Price, K. (1995) Differential Evolution—A Simple and Efficient Adaptive Scheme for Global Optimization over Continuous Spaces. International Computer Science Institute, Berkeley, CA, Technical Report TR-95-012.

[26] Arbex, R.O. and da Cunha, C.B. (2015) Efficient Transit Network Design and Frequencies

Setting Multi-Objective Optimization by Alternating Objective Genetic Algorithm. *Transportation Research Part B: Methodological*, **81**, 355-376.
http://dx.doi.org/10.1016/j.trb.2015.06.014

[27] Beasley, D., Bull, D.R. and Martin, R.R. (1993) An Overview of Genetic Algorithms: Part 2, Research Topics. *University Computing*, **15**, 170-181.
http://orca-mwe.cf.ac.uk/64433/1/ga_overview2.pdf

[28] Baaj, M.H. and Mahmassani, H.S. (1991) An AI-Based Approach for Transit Route System Planning and Design. *Journal of Advanced Transportation*, **25**, 187-209.
http://dx.doi.org/10.1002/atr.5670250205