Australasian Transport Research Forum 2013 Proceedings
2 - 4 October 2013, Brisbane, Australia
Publication website: http://www.patrec.org/atrf.aspx

A Swarm Based Method for Solving Transit Network Design Problem

Mehdi Bagherian*[1], Saghar Massah[2], and Shahab Kermanshahi[3]

[1] School of Civil Engineering, the University of Queensland, Australia.

[2] Islamic Azad University, Science and Research Branch, Tehran, Iran.

[3] Civil Engineering Department, Sharif University of Technology, Tehran, Iran

* Email for correspondence: m.bagherian@uq.edu.au

Abstract

In this study, a Discrete Particle Swarm Optimization (DPSO) algorithm is assimilated to solve the Transit Network Design Problem (TNDP). First, A Mixed Integer Model is developed for the TNDP. The solution methodology utilized here is made of two major elements. A route generation module is firstly developed to generate all the feasible transit lines. Through the second part, a DPSO algorithm is utilized to select the optimal set of lines from the constructed ones. The objective function is to maximize coverage index while satisfying the operator cost upper level constraints. The efficacy and accuracy of the implemented algorithms is compared with ones obtained by an enumeration process as well as an enumeration-based heuristic approach. Results confirmed that the PSO algorithm can find the optimum combination with significant decrease in the computational costs.

1. Introduction

Public transportation is known as a viable option for sustainable development of the transportation systems in urban areas. Improving the mobility, relieving the traffic congestion, its specific equity considerations, and noticeable reduction in fuel consumption and air pollution reduction is reported as the benefits of public transportation systems (Kepaptsoglou and Karlaftis, 2009). The problems of optimization of public transit systems are categorized into strategic, tactical, operational, and real-time control levels (Desaulniers and Hickman, 2007). The transit network design problem which is the focus of this paper is classified in the strategic level (Asadi Bagloee and Ceder, 2011).

The Transit Network Design Problem (TNDP) can be stated as designing a new transit network or to redesign (developing or modifying) an existing one (Ceder, 2007). This problem involves the minimization (or maximization) of a defined objective function. Numerous objective functions have been defined in the literature of this realm. Ceder (2007) classified the objective functions utilized for TNDP. Besides, Kepaptsoglou and Karlaftis (2009) performed a comprehensive review on the previous research and classified them based on their objective function, parameters, and solution methodology. Among the defined objective functions for the transit network design problem, maximization of the covered trips is the objective function of a handful of research. It urges the algorithm to select a set of rapid lines which provide a plausible serving time to as many demands as possible. Laporte et al. (2007) presented a mixed-integer model for Rapid Transit Network Design Problem (RTNDP) where the trip coverage were defined as the objective function. Marín (2007) then presented the extended model of the previously presented one. In their new formulation, the number of lines assumed to be a variable. Besides, the definition of origins and destinations for the lines were relaxed. Escudero and Muñoz (2009) developed a two-stage method to solve RTND where a connected graph were firstly resulted and the transit routes were then extracted from that graph. Kermanshahi et al. (2010) studied RTND and presented a new model and solution algorithm that considered the transfer penalty. Maximization of the trip coverage is what we utilized in the present study.

Swarm intelligence is a branch of artificial intelligence which is based on study of individuals' behavior in various decentralized systems (Teodorovic, 2008). In this realm, a swarm is defined as a group of agents which communicate with each other in a defined region (searching space) to find the optimal solution. Different interaction methodologies have resulted in the emergence of a variety of problem-solving approaches over the past years. Particle Swarm Optimization (Kennedy and Eberhart, 1995), Ant Colony Optimization (Dorigo et al., 2006), Stochastic Diffusion Search (Bishop, 2007), and Bee Colony Optimization (Teodorović and Dell'Orco, 2005) can be mentioned as the most known swarm based algorithms. Among them, Ant Colony Optimization has absorbed researcher's attention for tackling complex transportation problems such as vehicle routing and scheduling, public transit, traffic engineering, as well as control problems(Teodorovic, 2008). Particle Swarm Optimization, on the other hand, in spite of its robust potential upon complex engineering problems, is rarely applied to transportation problems. Application of Discrete Particle Swarm Optimization(Kennedy and Eberhart, 1997) to vehicle routing problem (XIAO et al., 2005) (XIAO et al., 2005) and network design problem (Babazadeh et al., 2011) can be found in the literature.

In this study, a Discrete Particle Swarm Optimization (DPSO) algorithm is utilized for solving the Transit Network Design Problem (TNDP). The solution methodology utilized here is made of two major elements. First, a route generation module is developed to generate all the feasible transit lines. Through the second part, a DPSO algorithm is utilized to select the optimal set of lines from the constructed lines. The objective function is to maximize coverage index while satisfying the operator cost upper level constraint. The efficacy and accuracy of the implemented algorithms is compared with ones obtained by an enumeration process as well as an enumeration-based heuristic approach.

The subsequent sections of this paper are organized as follows. The next section is devoted to definition of the transit network design problem. Particle Swarm Optimization method is described in the fourth section. Then a case study example is solved and the results are compared with what we obtained using exhaustive and modified enumeration methods.

1. PROBLEM FORMULATION

The Transit Network Design Problem can be stated as a discrete combinatorial optimization problem: The selection of a pre-specified number of transit routes from a given set of candidate paths. The objective function is to maximize the defined coverage index while satisfying the budget upper bound constraints. The coverage index of a possible network is the summation of all the covered demand in the network. To obtain this value, it is assumed that the travel time for each origin-destination (named here as Target Travel Time) is given and known. It may be valuable to study passenger preferences to generate a target travel matrix, probably for different passenger market segments. For each possible set of transit lines, the transit travel time for each trip would be calculated. If this value is smaller than the corresponding target travel time, the trip is assumed to be covered. It is noteworthy to mention that the main inputs of the model are the set of candidate routes, demand matrix, and Target Travel Time matrix.

The objective function utilized in this study is the generalized definition of Laporte et al. (2007). To calculate the transit travel time, all the elements of a trip (access time, egress time, waiting time, transfer penalty, and in-vehicle time) are considered. The details of the calculations are presented through the previous work of the authors (Kermanshahi, 2012) thus not presented in this paper. The mathematical formulation of the problem can be presented as follows:

$$maximize\ Z = \sum_{w=1}^{W} c^w \times g^w \qquad (1)$$

$$\sum_{l=1}^{L} y^l \times c^l \leq B \qquad (2)$$

$$x_{ij}^{l} \leq \delta_{ij}^{l} \times y^{l} \times M \qquad \forall ij \in E, \forall l \in L \qquad (3)$$

$$T^{w} - c^{w} \times t^{*w} + M \times (1 - c^{w}) \geq 0 \quad \forall w \in W \qquad (4)$$

When

c^{w}	binary variable of coverage, which is 1 if w is covered and 0, otherwise
g^{w}	number of trips between w OD pair
y^{l}	binary variable of selecting route l which is 1 if route l is selected and 0, otherwise
c^{l}	construction cost of route l
B	budget level
$x_{\mathrm{ij}}^{\mathrm{l}}$	If link ij belongs to route l and route l is selected in the solution.
$\delta_{\mathrm{ij}}^{\mathrm{l}}$	is 1 if link ij belongs to route l and 0, otherwise
T^{w}	target travel time between w OD pair
$\mathrm{t}^{*\mathrm{w}}$	transit travel time, obtained from the transit assignment procedure
M	a big enough number

Equation (1) represents the objective function that is the summation of covered trips over all OD pairs. Constraint (2) stands to guarantee that the solution is budget feasible. The construction cost of each single route is calculated at route generation procedure. Constraints (3) guarantee that if a route is not selected, it is not used by transit passengers. Constraints (4) reflect the coverage definition for an OD pair w; if the provided service by transit system ($\mathrm{t}^{*\mathrm{w}}$) is greater than c^{w} (the target travel time of *w*), then the corresponding coverage binary variable, is forced to be zero. The procedure of route generation procedure as well as the transit assignment is comprehensibly described in (Kermanshahi, 2012). Some of the major constraints of BRT route layout and passengers movement in the transit network are not presented in this formulation. Indeed, the constraints and characteristics of the transit lines are implicitly considered in the route generation module.

2. SOLUTION METHODOLOGY

In this study, we utilized a Discrete PSO algorithm for solving the transit network design problem. Upon the previous work of the authors, the problem was solved using an enumeration based method. Here we assimilated a PSO method to select a set of candidate paths which maximizes the coverage index while the budget limitation constraint is not violated. The variables of the problem are the indexes of the lines. Furthermore, the dimension of the problem is maximum number of lines that can be added to the current system. Since this value is not known, one plausible strategy is to solve the problem for different dimensions. Here we used the values obtained from the enumeration-based methods, developed by Kermanshahi (2012).

2.2 Particle Swarm Optimization

Inspired by the social behavior of bird flocking and fish schooling, particle swarm optimization is an evolutionary computation model that has its roots in artificial life. First proposed by Kennedy and Eberhart (1995) , PSO performs a swarm-based search using particles to represent potential solutions within the search space. Each particle is characterized by its position, velocity, and a record of its past performance.

In the basic PSO algorithm, using the following equations, the position (x) and velocity (v) of each particle in the swarm can be updated upon each iteration.

$$x(t+1) = x(t) + v(t+1) \qquad (5)$$

$$v(t+1) = v(t) + C_1 r_1(t)[g(t) - x(t)] + \; C_2 r_2(t)[p(t) - x(t)] \qquad (6)$$

Where C_1 and C_2 are the accelerator constants, r_1 and r_2 are randomly generated numbers that distribute uniformly in the [0,1] interval, *g(t)* is the best answer found by the population to that point, and *p(t)* is the best answer found by each particle.

There are many extensions of basic PSO to improve its convergence behavior.Shi and Eberhart (1998) introduced the "inertia weight model" in which the inertia of the particle (the *v(t)* term) would be multiplied by a *w* parameter. This parameter plays an important role in trading off the exploration and exploitation of the algorithm. Large amounts of inertia weight will improve the exploration, but the diversity may also increase. In contrast, small values of w increase the possibility of searching in a specific area to obtain better solutions but may also increase the probability of trapping in local optimums (Parsopoulos and Vrahatis, 2010). Consequently, equation (6) can be rewritten as:

$$v(t+1) = \boldsymbol{w} v(t) + C_1 r_1(t)[g(t) - x(t)] + C_2 r_2(t)[p(t) - x(t)] \qquad (7)$$

In this equation, the inertia weight can either stay constant or decrease during the iterations of the algorithm. A dynamic value for the inertia weight is utilized in this study.

Another modification is velocity clamping (Eberhart et al., 1996). In each iteration, if the calculated displacement of a particle exceeds the specified maximum velocity, it is set to the maximum value. Let $V_{max,j}$ denote the maximum velocity allowed in dimension j. The particle velocity is then adjusted before the position update using.

$$v_{ij}(t+1) = \begin{cases} v'_{ij}(t+1) & if\ v'_{ij}(t+1) \le V_{max,j} \\ V_{max,j} & if\ v'_{ij}(t+1) \ge V_{max,j} \end{cases} \qquad (8)$$

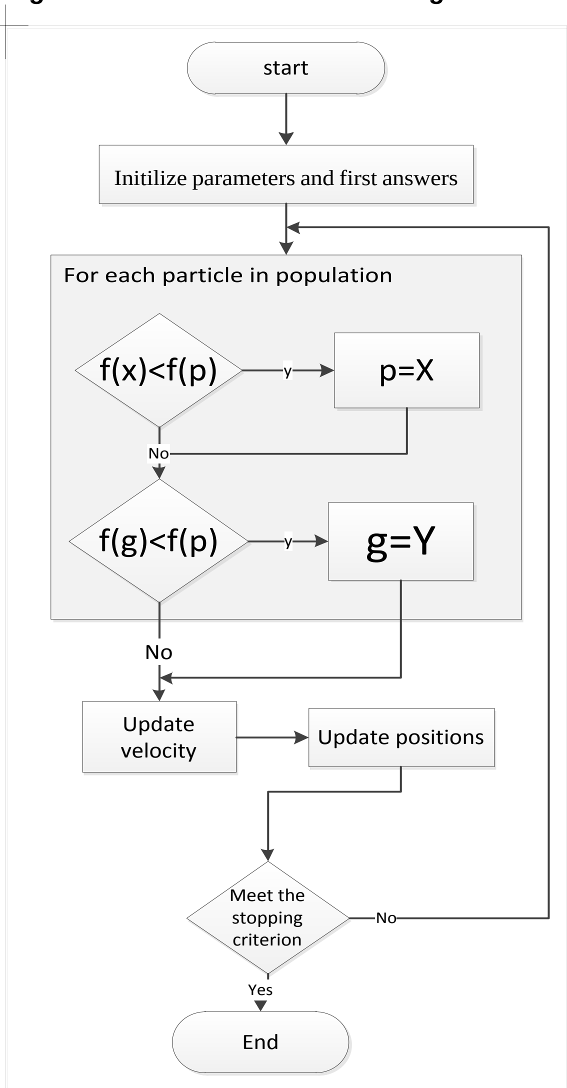
Since the basic PSO is developed for tackling problems with continuous domain, it should be modified for solving the addressed TNDP which is discrete problem. This modifications is performed using a simple converting from real values to integer ones, applied on each single term of the equation (7). In other words, the velocity updating equation is modified as follows:

$$v(t+1) = \boldsymbol{int}(wv(t)) + \boldsymbol{int}(C_1 r_1(t))\,[g(t) - x(t)] + \boldsymbol{int}(C_2 r_2(t))[p(t) - x(t)] \qquad (9)$$

Once the terms were converted to integer values, the velocity of the particles in each step would be integer as well. Consequently, PSO is forced to search only the integer values (i.e. the index of each candidate path) as the variables.

Figure 1 exhibits the flowchart of the PSO algorithm which we utilized for tackling the transit network design problem.

Figure 1: Flowchart of the PSO Algorithm

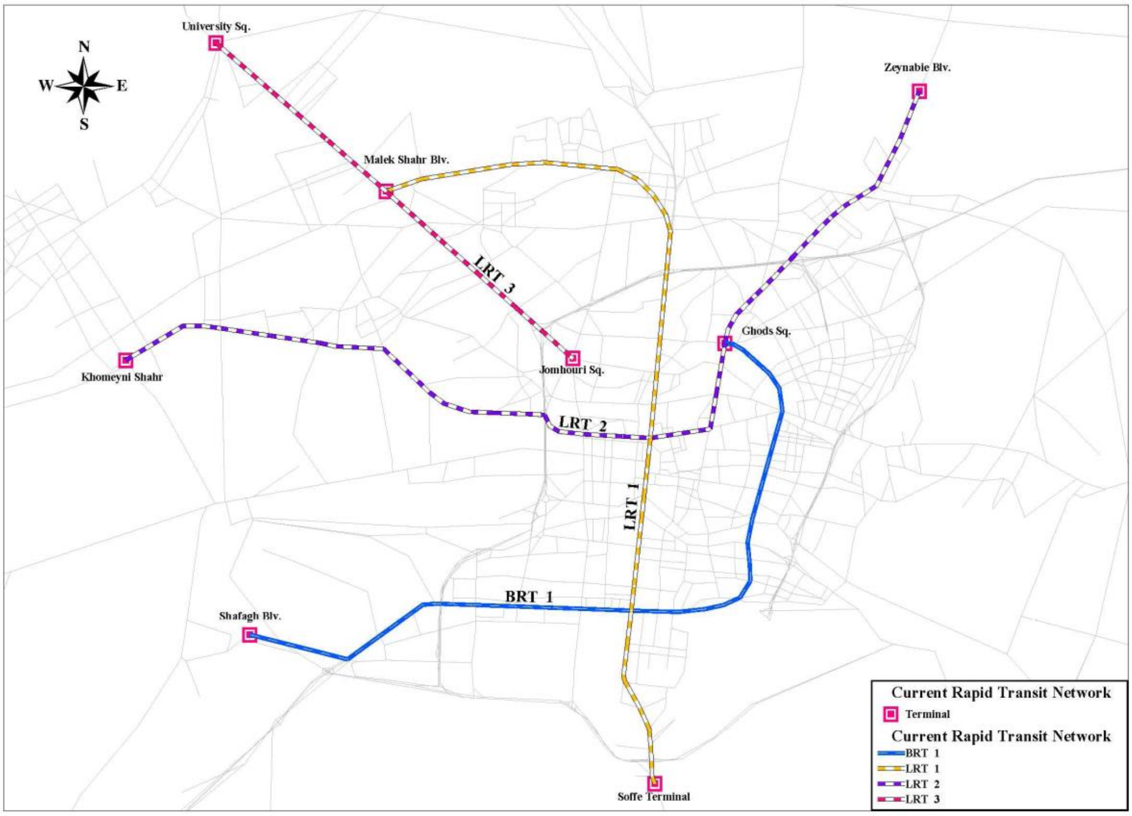


3. NUMERICAL EXAMPLE

The software was developed in the .Net environment using the C# programming language. All the runs were performed on a laptop computer with an Intel® Core™ 2 Dual 2.80-GHz processor and 4 GB of installed memory (RAM). The calculation of objective function and the generation of initial answers were performed using a unique module to make the runs of the algorithms analogous. The stopping criterion is assigned to be met when no improvement is observed over a predefined number of iterations.

The model and solution algorithm was applied to the Isfahan metropolitan road network. The network is consisted of a set of available rapid transit lines. Figure 2 shows the Isfahan metropolitan road network and the existing transit lines. The lines are either constructed or approved. Our goal is to extend rapid transit network using BRT lines.

Figure 2: Isfahan Metropolitan Area and the existing Transit Lines



The inputs of the model are classified as threefold. 1) BRT system specifications; 2) Network information; and 3) demand information. Further details for each single input is presented at (Kermanshahi, 2012) and are not replicated in this paper. The Input data files are also available at www.mehdibagherian.com/GreaterIsfahanData. The characteristics of the network as well as the parameters assumed for this example are summarized in table 1.

Table 1: Network Characteristics and parameters assumed for the case study

Parameter	Value
Total Number of Links	129
Total Number of Nodes	113
Number of Terminal Nodes	14
Total Number of Trips	2148000
Covered Trips in existing Network	1357006
BRT Speed(Km/h)	30
Minimum Length of Transit Lines(Km)	8
Maximum Length of Transit Lines(Km)	15
Candidate Paths	56

Upon the previous works, We solved the problem for different upper level of budgets using an enumeration method as well as a enumeration-based heuristic approach(Kermanshahi, 2012). In other words, all the possible combinations were evaluated and the ones which resulted the maximum coverage while satisfied the constraint were identified. The results are utilized here as the benchmark and the results obtained by the PSO algorithm were compared to evaluate the efficiency of this swarm based approach.

3.1 Tuning of the PSO Algorithm

PSO has a set of parameters which need to be tuned, including the swarm size, inertia weight, and accelerator coefficients. Although some recommendations are available in the literature for selection of optimum parameters, they are problem dependent and should be calibrated for each single problem. In this study, we performed a sensitivity analysis on the parameters of the algorithm to address the trend of the PSO algorithm. Unless otherwise specified, the following values are taken for the PSO parameters: Swarm Size = 60, $C_1=C_2=2$, $V_{max}=0.4$ of the range of variables upon each single iteration, and a dynamic Inertia Weight (*w*), changing from 0.9 to 0.1 over the searching process. These values are selected according to the values suggested in the literature of PSO method. For each single scenario, ten different runs were performed.

Table 2 demonstrates the impact of the swarm size on the objective function and number of evaluations. More particles of the swarm cause the better diversity of the swarm and more appropriate coverage of the searching space. However, increasing the swarm size causes further computational costs and the searching process degrades to a parallel random search(Engelbrecht, 2007). It can be seen that the swarm sizes of 60, 70, and 80 caused the best answers without trapping in local optimums through the performed runs. Among them, the efficiency of the algorithm (the speed of convergence) for swarm size=60 is better than the other two amounts. To present the efficiency of each single algorithm in reaching the optimal answer, the percentage of error index is defined using the following equation:

$$Error = \frac{C(Best\ Solution) - C(Algorithm)}{C(Best\ Solution)} \qquad (10)$$

Where *C (Algorithm)* =the best objective function found by the algorithm and *C (Best Solution)* = the optimum solution of the algorithm, found by enumeration method.

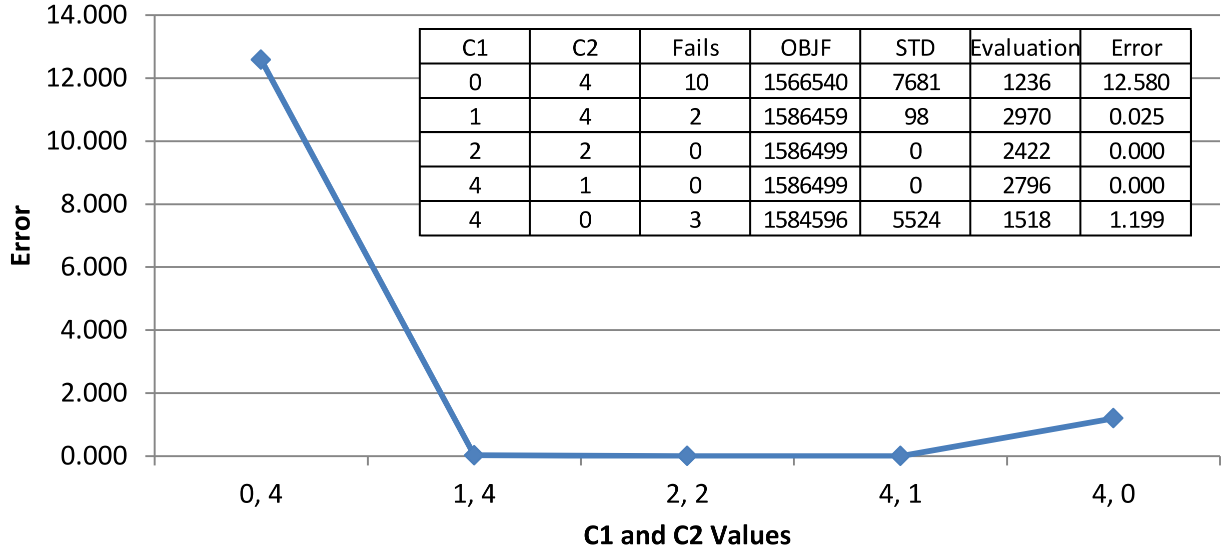
Table 2: Sensitivity analysis on the swarm size

10	8	1584560	1.223	2719	792
20	7	1585087	0.890	2525	1116
30	4	1586387	0.070	137	1620
40	3	1585916	0.368	1568	1208
50	3	1586415	0.053	128	1725
60	0	1586499	0.000	0	2422
70	0	1586499	0.000	0	2891
80	0	1586499	0.000	0	2672
90	3	1586415	0.053	128	3195

Five different sets of accelerator coefficient values were observed to determine which would lead to a better solution. The acceleration coefficients, C_1 and C_2, together with the random vectors, r_1 and r_2, control the stochastic influence of the cognitive and social components on the overall velocity of a particle. While C_1 controls the effect of the best solution found by each particle, the C_2 value is the weight of the best solution found by the swarm so far. Figure 2 demonstrates the results obtained for different values of the accelerator coefficient. Results show that the PSO algorithm could find the optimum solution in three defined scenarios.

However, the cognitive and social only models (C_1=4, C_2=0 and C_1=0, C_2=4, respectively) resulted in inferior answers.

Figure 3 Sensitivity Analysis on the Accelerator Coefficients (C1,C2)



C1	C2	Fails	OBJF	STD	Evaluation	Error
0	4	10	1566540	7681	1236	12.580
1	4	2	1586459	98	2970	0.025
2	2	0	1586499	0	2422	0.000
4	1	0	1586499	0	2796	0.000
4	0	3	1584596	5524	1518	1.199

4. Computational Results

We solved the TNDP for the Isfahan Network for different levels of available budget. Table 3 summarizes the results obtained for different scenarios. As shown in the table, the increase in objective function is commensurate with the budget level. Furthermore, since the construction cost is the function of total length of proposed lines, the number of lines in the optimum combination increases with augmentation in the available budget. The characteristic of each single combination is summarized in the table.

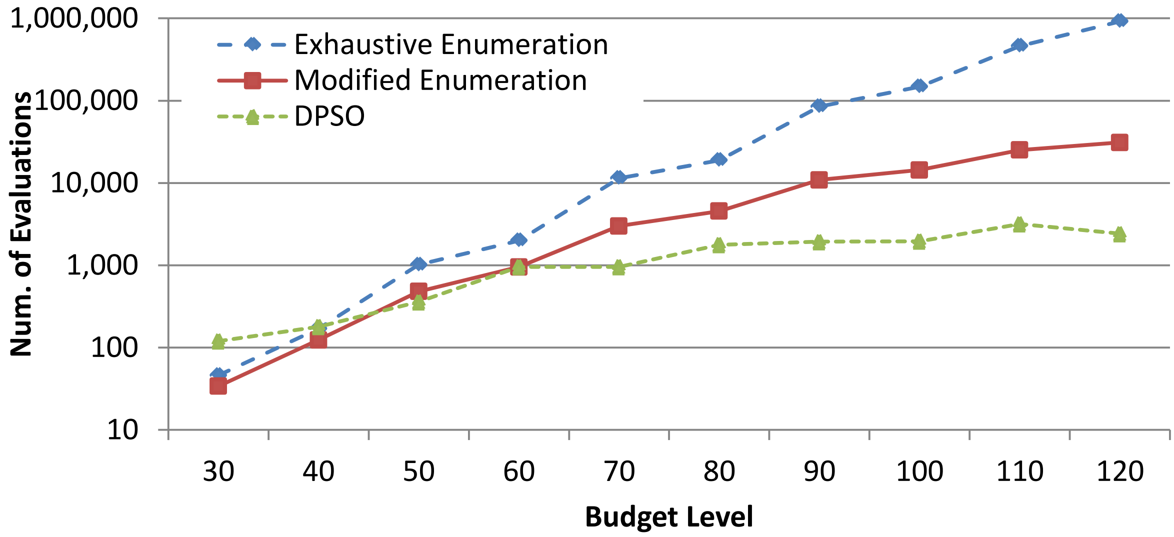
The results obtained by three different solution methodologies (i.e. exhaustive enumeration, an enumeration-based heuristic approach, and the DPSO algorithm) are also shown in this table. Comparing the required evaluations for DPSO with the results of the prior research (Kermanshahi, 2012), the optimum solution is acquired with a significant decrease in the performed objective function calculations. Indeed, On the contrary of the number of evaluations performed in enumeration-based methods, we can observe a slight change in calculation cost while increasing the level of budget. This trend is exponentially changing with increase in budget for the previously used solution methodologies.

Figure 4 shows the required number of evaluations to find the optimum combination for different budgets. For the lower available budgets, the computational cost in enumeration-based methods is comparable with the PSO-based one's. In these scenarios, since the number of possible combinations is limited and enumerable, using the exact methods seems plausible. However, the algorithms' efficiency is disparate in higher budgets. For example, comparing the results for the budget of USD 120 million, PSO could find the answer with only 2422 evaluations which is almost 13 times faster than the method presented in (Kermanshahi, 2012) and 380 times faster than performing an exhaustive enumeration method. The significant achieved value in saving the computational costs is remarkable while solving the problems with larger feasible searching space where the enumeration-based methods are no longer efficient.

Table 3 : Optimization results, obtained for different levels of available budget

Budget (million USD)	**Objective function**	**Optimal combination**	**Coverage Ratio**	**Number of Evaluations**		
				Exhaustive Enumeration	Modified Enumeration	DPSO
30	1444856	*Route 5*: 6-9-22-24-26-12-13-14-15	67.2	46	34	120
60	1499122	*Route 5*: 6-9-22-24-26-12-13-14-15 *Route 17*: 2-15-14-30-28-16	69.7	1993	956	960
90	1558164	*Route 17*: 2-15-14-30-28-16 *Route 23*: 5-7-10-9 *Route 25*: 6-9-22-24-26-12-13-14 *Route 38*: 5-7-11	72.5	85038	10870	1938
120	1586499	*Route 5*: 6-9-22-24-26-12-13-14-15 *Route 9*: 5-7-11-18-17 *Route 17*: 2-15-14-30-28-16 *Route 23*: 5-7-10-9 *Route 27*: 3-1-20-17-18-11	73.8	919038	31077	2422

Figure 4 Number of Evaluations for obtaining the Optimal Solution



5. CONCLUSION

In this study, the efficiency of a Discrete Particle Swarm Optimization algorithm was observed for solving the Transit Network Design Problem. The DPSO algorithm was assimilated to find the optimal set of lines from the constructed lines. The objective function was defined as maximization of the coverage index while satisfying the upper level of budget constraints. The solution method was utilized for solving the problem in different level of budget. In all the scenarios, the optimum solution was found upon a plausible number of evaluations. The required number of evaluations to reach the optimum combination was compared with the results obtained by an exhaustive enumeration method as well as an enumeration-based heuristic approach. The results confirmed the capability of the DPSO to find the optimum solution with a significant decrease in the total computational cost.

6. References

ASADI BAGLOEE, S. & CEDER, A. A. 2011. Transit-network design methodology for actual-size road networks. *Transportation Research Part B: Methodological*.

BABAZADEH, A., POORZAHEDY, H. & NIKOOSOKHAN, S. 2011. Application of particle swarm optimization to transportation network design problem. *Journal of King Saud University-Science*.

BISHOP, J. M. 2007. Stochastic diffusion search. *Scholarpedia,* 2**,** 3101.

CEDER, A. 2007. *Public transit planning and operation: theory, modeling and practice*, Elsevier, Butterworth-Heinemann, Oxford, UK.

DESAULNIERS, G. & HICKMAN, M. D. 2007. Public transit. *In:* BARNHART, C. & G., L. (eds.) *Handbook in OR & MS.* Elsevier.

DORIGO, M., BIRATTARI, M. & STUTZLE, T. 2006. Ant colony optimization. *Computational Intelligence Magazine, IEEE,* 1**,** 28-39.

EBERHART, R. C., SIMPSON, P. K., DOBBINS, R. & DOBBINS, R. W. 1996. *Computational intelligence PC tools*, AP Professional.

ENGELBRECHT, A. P. 2007. *Computational intelligence: an introduction*, Wiley.

ESCUDERO, L. & MUÑOZ, S. 2009. An approach for solving a modification of the extended rapid transit network design problem. *Top,* 17**,** 320-334.

KENNEDY, J. & EBERHART, R. Particle swarm optimization. 1995. IEEE, 1942-1948 vol. 4.

KENNEDY, J. & EBERHART, R. C. A discrete binary version of the particle swarm algorithm. 1997. IEEE, 4104-4108 vol. 5.

KEPAPTSOGLOU, K. & KARLAFTIS, M. 2009. Transit route network design problem: Review. *Journal of transportation engineering,* 135**,** 491.

KERMANSHAHI, S. A., SHAFAHI, Y., ZANGUI, M. & MOLLANEJAD, M. Rapid transit network design: a new model and solution algorithm. TRB2010 Annual Meeting, 2010 Washigton DC.

KERMANSHAHI, S. A., SHAFAHI,Y., BAGHERIAN,M. 2012. Application of a New Rapid Transit Network Design Model to BRT Network Design. *Transport,* Accepted for Publication.

LAPORTE, G., MARÍN, Á., MESA, J. & ORTEGA, F. 2007. An integrated methodology for the rapid transit network design problem. *Algorithmic Methods for Railway Optimization***,** 187-199.

MARÍN, Á. 2007. An extension to rapid transit network design problem. *Top,* 15**,** 231-241.

PARSOPOULOS, K. E. & VRAHATIS, M. N. 2010. *Particle swarm optimization and intelligence: advances and applications*, Information Science Reference.

SHI, Y. & EBERHART, R. A modified particle swarm optimizer. 1998. IEEE, 69-73.

TEODOROVIC, D. 2008. Swarm intelligence systems for transportation engineering: Principles and applications. *Transportation Research Part C: Emerging Technologies,* 16**,** 651-667.

TEODOROVIĆ, D. & DELL'ORCO, M. Bee colony optimization–a cooperative learning approach to complex transportation problems. 2005. 51-60.

XIAO, J., HUANG, Y., LI, J. & WANG, X. 2005. Vehicle Routing Problem Based on Discrete Particle Swarm Optimization [J]. *Systems Engineering,* 4**,** 97—100.