Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia - Social and Behavioral Sciences 00 (2013) 000–000

Procedia
Social and Behavioral Sciences

www.elsevier.com/locate/procedia

EWGT2013 – 16th Meeting of the EURO Working Group on Transportation

Modified differential evolution algorithm for the continuous network design problem

Ozgur Baskan[a,]*, Huseyin Ceylan[a]

[a] *Department of Civil Engineering, Engineering Faculty, Pamukkale University, Kinikli Campus, 20070, Denizli, Turkey*

Abstract

The Continuous Network Design Problem (CNDP) is recognized to be one of the most difficult problems in transportation field since the bilevel formulation of the CNDP is nonconvex. On the other hand, the computation time is crucial importance for solving the CNDP because the algorithms implemented on real sized networks require solving traffic assignment model many times, which is the most time consuming part of the solution process. Although the methods developed so far are capable of solving the CNDP, an efficient algorithm, which is able to solve the CNDP with less number of User Equilibrium (UE) assignments, is still needed. Therefore, this paper deals with solving the CNDP using MOdified Differential Evolution (MODE) algorithm with a new local search and mutation operators. For this purpose, a bilevel model is proposed, in which the upper level problem deals with minimizing the sum of total travel time and investment cost of link capacity expansions, while at the lower level problem, UE link flows are determined by Wardrop's first principle. A numerical example is presented to compare the proposed MODE algorithm with some existing methods. Results showed that the proposed algorithm may effectively be used in order to reduce the number of UE assignments in solving the CNDP.

Keywords: Continuous network design problem; differential evolution; bilevel programming

1. Introduction

A Continuous Network Design Problem (CNDP) is to determine the set of link capacity expansions and the corresponding equilibrium link flows for which the measure of performance index for the network is optimal (Chiou, 2005). The CNDP can be described as one of the most computationally intensive problem in transportation field. Since the multiple objectives exist in the CNDP, it can be formulated as bilevel programming

* Corresponding author. Tel.: +90-258-2963416; fax: +90-258-2963460.
E-mail address: obaskan@pau.edu.tr

model which is difficult to solve. The difficulty stems from solving the lower level problem for each feasible set of upper level decision variables. In the CNDP, upper level objective function can be defined as the sum of total travel time and investment cost of link capacity expansions whilst the lower level is formulated as traffic assignment model which can be static or dynamic.

The first formulation for solving the CNDP was proposed by Abdulaal and LeBlanc (1979) using Hooke-Jeeves (HJ) heuristic algorithm. Tan et al. (1979) attempted to solve the CNDP by expressing the traffic equilibrium problem by a set of nonlinear and nonconvex, but differentiable constraints in terms of path flow variables. Suwansirikul et al. (1987) developed Equilibrium Decomposition Optimization (EDO) algorithm for solving the CNDP. Marcotte and Marquis (1992) used an efficient heuristic procedure for solving the CNDP. In addition, sensitivity-based heuristic algorithms were developed for the CNDP in different studies (Cho, 1988; Friesz et al., 1990; Yang, 1995; 1997). Furthermore, Friesz et al. (1992) used Simulated Annealing (SA) approach to solve the CNDP. Their results showed that the proposed heuristic is more efficient than Iterative Optimization Assignment (IOA) algorithm, HJ algorithm and EDO approach. Davis (1994) used the generalized reduced gradient method and sequential quadratic programming to solve the CNDP. Meng et al. (2001) used single level model in solving the CNDP in order to avoid the disadvantages of the bilevel programming model. Chiou (2005) proposed gradient based methods to solve the CNDP and achieved good results especially when the congested road networks are considered. Similarly, Ban et al. (2006) presented a relaxation method to solve the CNDP when the lower level is a nonlinear complementary problem. Karoonsoontawong and Waller (2006) proposed SA, Genetic Algorithm (GA), and random search techniques to solve the mentioned problem. Their study showed that GA performed better than the others on the test networks. Xu et al. (2009) used SA and GA algorithms to find optimal solutions of the CNDP. They emphasized that quality of the results are dependent on the demand level. Recently, Wang and Lo (2010) proposed an approximation method for finding the globally optimal solution of the CNDP and Li et al. (2012) presented a viable global optimization method for the CNDP.

Up to date, studies have been focused on the CNDP are generally based on the heuristic approaches, and guarantee finding a solution which is at least locally optimal. Nevertheless, an efficient heuristic algorithm, which is capable of solving the CNDP with less number of User Equilibrium (UE) assignments, is still needed. Therefore, this paper deals with determining the optimal link capacity expansions for a given road network using Modified Differential Evolution (MODE) algorithm.

The rest of this paper is organized as follows. In Section 2, notation for the classical CNDP is given. The problem formulation is summarized in Section 3. In the next section, proposed solution method is presented. In Section 5, numerical application is conducted on example test network. Conclusions are drawn in Section 6.

2. Notation

A	the set of links
K_{rs}	the set of paths between O-D pair $rs \; \forall r \in R,\ s \in S$
R	the set of origins
S	the set of destinations
$\mathbf{D}$	the vector of O-D pair demands, $\mathbf{D} = [D_{rs}] \; \forall r \in R,\ s \in S$
$\mathbf{f}$	the vector of path flows, $\mathbf{f} = [f_k^{rs}], \forall r \in R,\ s \in S,\ k \in K_{rs}$
$\mathbf{t}$	the vector of link travel times, $\mathbf{t} = [t_a(x_a, y_a)] \; \forall a \in A$
$\mathbf{u}$	the vector of upper bound for link capacity expansions, $\mathbf{u} = [u_a], \forall a \in A$
$\mathbf{x}$	the vector of equilibrium link flows, $\mathbf{x} = [x_a], \forall a \in A$
$\mathbf{y}$	the vector of link capacity expansions, $\mathbf{y} = [y_a], \forall a \in A$

d_a the cost coefficient, $\forall a \in A$

θ_a the link capacity, $\forall a \in A$

Z upper level objective function

z lower level objective function

ρ the conversion factor from investment cost to travel times

$g_a(y_a)$ the investment function, $\forall a \in A$

$\delta_{a,k}^{rs}$ the link/path incidence matrix variables, $\forall r \in R,\ s \in S, k \in K_{rs}, a \in A$. $\delta_{a,k}^{rs} = 1$ if route *k* between O-D pair *rs* uses link *a*, and $\delta_{a,k}^{rs} = 0$ otherwise

α_a , β_a the parameters of link cost function, $\forall a \in A$

3. Problem formulation

The CNDP can be formulated as follows:

$$\min_{\mathbf{x},\mathbf{y}} Z(\mathbf{x},\mathbf{y}) = \sum_{a \in A} (t_a(x_a, y_a)x_a + \rho g_a(y_a)) \quad (1)$$

$$\text{s.t.} \quad 0 \le y_a \le u_a \ , \quad \forall a \in A \quad (2)$$

where $\mathbf{x}(\mathbf{y})$ is vector of the equilibrium link flows which is the solution of the following convex optimization problem:

$$\min_{\mathbf{x}} \ z = \sum_{a \in A} \int_0^{x_a} t_a(w, y_a)dw \quad (3)$$

$$\text{s.t.} \quad \sum_{k \in K} f_k^{rs} = D_{rs} \quad \forall r \in R,\ s \in S,\ k \in K_{rs} \quad (4)$$

$$x_a = \sum_{rs} \sum_{k \in K_{rs}} f_k^{rs} \delta_{a,k}^{rs} \quad \forall r \in R,\ s \in S,\ a \in A, k \in K_{rs} \quad (5)$$

$$f_k^{rs} \ge 0 \quad \forall r \in R,\ s \in S,\ k \in K_{rs} \quad (6)$$

In general, the users' route choice is characterized by the UE assignment in solving the CNDP. The UE assignment problem is to find the link flows, **x**, which satisfies the user-equilibrium criterion when all the Origin-Destination (O-D) demands have been assigned. On the other hand, link capacity expansions without considering the response of road users may actually increase level of congestion on the road network. Therefore, prediction of traffic flows is crucial importance in solving the CNDP.

4. Solution method

4.1. Basics of DE

The Differential Evolution (DE) is a simple and powerful algorithm which is introduced by Storn and Price (1995) to solve various optimization problems. The DE guides the initial population towards to vicinity of the global or near-global optimum solution for a given optimization problem through repeated cycle of mutation, crossover and selection (Liu et al., 2010). In the DE, three control parameters are used to conduct the optimization process. One of them is the number of populations (NP) that represents the number of solution vectors used in the solution process. The second one is the mutation factor (F), which is used to obtain mutant vector from selected three solution vectors in the population and recommended to set between 0.5-1 by Storn and Price (1995). The last one is the crossover rate (CR) that is the probability of consideration of the mutant vector. The recommended range of the crossover rate is [0.8, 1] by Storn and Price (1995). In this study, F and CR are selected as 0.8 for all numerical applications. The procedure of the DE algorithm can be summarized as follows. Note that we preferred to describe the solution process of the DE combining with the CNDP in order to provide brevity of the paper.

Generation of the initial population: At generation *t*, the initial solution vector, $\mathbf{y}^t = \left\{ y_i^{j,t}, y_i^{j,t}, ..., y_i^{j,t} \right\}$, where $i \in \{1, 2, ..., N\}$ and $j \in \{1, 2, ..., NP\}$, is filled with randomly generated capacity expansions for a set of selected links in a given road network by considering upper and lower boundaries. Then their corresponding objective function values are calculated using Eqs. (1-6). Note that *N* represents the number of set of selected links on road network.

Mutation: The mutation is performed by adding the weighted difference vector between two solution vectors to a third vector. A mutant vector, $\mathbf{m}^t = \left\{ m_i^{j,t}, m_i^{j,t}, ..., m_i^{j,t} \right\}$, can be created as shown in Eq. (7).

$$m_i^{j,t} = y_i^{1,t} + F(y_i^{2,t} - y_i^{3,t}) \tag{7}$$

where $y_i^{1,t}$, $y_i^{2,t}$ and $y_i^{3,t}$ are randomly selected capacity expansions within the range [0,NP] at generation *t*, and $y_i^{1,t} \neq y_i^{2,t} \neq y_i^{3,t}$.

Crossover: The search process of the DE is completed with the crossover operator. At this step, each member of the trial vector, $r_i^{j,t}$, is chosen from the mutant vector with the probability of CR or from the target vector with the probability of (1-CR) as given in Eq. (8).

$$r_i^{j,t} = \begin{cases} m_i^{j,t}, & \text{if rand } (0,1) \leq \; CR \quad \text{or } i = i_{rand} \\ y_i^{j,t}, & \text{otherwise} \end{cases} \tag{8}$$

As can be seen from Eq. (8), CR is compared with the output of a uniform random number generator, $\text{rand}\,(0,1)$, to determine either mutant vector or target vector will provide the member of the trial vector. If the random generated number is less than or equal to CR at generation *t*, the trial parameter is chosen from the mutant vector, $\mathbf{m}^{j,t}$; otherwise the parameter is chosen from the target vector, $\mathbf{y}^{j,t}$. Additionally, the constraint, $i = i_{rand}$, where i_{rand} is the uniformly distributed random number in the range [1,*N*], ensures that at least one member of the trial vector is taken from the mutant vector.

Selection: At this step, the trial vector, $\mathbf{r}^t$, is compared with the parent individual $\mathbf{y}^t$ by way of determining the objective function values and the best one enters to the generation t+1.

$$\mathbf{y}^{t+1} = \begin{cases} \mathbf{r}^t, & \text{if } f(\mathbf{r}^t) \leq f(\mathbf{y}^t) \\ \mathbf{y}^t, & \text{otherwise} \end{cases} \tag{9}$$

Termination: Mutation, crossover and selection steps of the DE are repeated until a predetermined stopping criterion is met or maximum number of generations (MGN) is reached.

4.2. Improvement mechanisms

4.2.1. Improvement 1: Although the DE algorithm is a simple and powerful heuristic algorithm, it can be further improved to solve complex optimization problems. For this purpose, we propose the MODE algorithm with newly developed local search and mutation operators. First improvement on the standard DE is performed by way of taking different mutation strategies into account. Thus, we defined a new parameter called Mutation Strategy Consideration Rate (MSCR) to decide which strategy is used at the mutation process of the MODE. The basic rule for selecting the mutation strategy is given as follows:

$$m_i^{j,t} = \begin{cases} y_i^{1,t} + F(y_i^{2,t} - y_i^{3,t}), & \text{if rand } (0,1) < \text{MSCR} \\ y_i^{1,t} + F(y_i^{best,t-1} - y_i^{2,t}), & \text{otherwise} \end{cases} \tag{10}$$

The proposed mutation process differs from the base DE in two aspects: one is that we use two mutation strategies simultaneous with MSCR parameter, and the other one is that we take the effect of best solution vector determined at previous generation into account with probability (1-MSCR). Thus, the MODE has ability to reach faster to global or near global optimum solutions. The recommended range of the parameter of MSCR is [0.9, 1]. This range has been determined after several experiments made by authors with different value of MSCR. It should be emphasized that this parameter setting is crucial importance because low values of MSCR may lead to the premature convergence of the MODE. On the contrary, its high values may produce unacceptable solutions especially for complex optimization problems. In this study, we used the value of 0.95 of MSCR for solving the CNDP.

4.2.2. Improvement 2: The second improvement mechanism is a kind of embedding a local search mechanism within the DE algorithm. Because of the DE's stochastic nature, it may suffer from slow convergence to the specified problem. On the other hand, increasing the rate of convergence of the algorithm may increase the chance of getting trapped to the local optima. It is clear that one of the possible ways to increase the rate of convergence is to save the best solution vector at each generation. Thus, we use local search method in order to improve the fitness value of the best solution vector that produced at each generation. In other words, our goal with this improvement is to push the best solution existed in the population towards to the global or near global optimum one step closer. The pseudo-code of the MODE algorithm with proposed improvements is shown in Fig. 1. In the local search process, the MODE algorithm generates dx vector from [α_1,α_2] range to adjust the length of jump for the best solution vector at each generation. The values of α_1 and α_2 are selected according to the upper and lower bounds of link capacity expansions, and decreased after local search phase is ended in order to reduce the search space around the best solution step by step. If the relative error between the average and best objective function values in the memory is less than a predetermined value, the MODE is terminated.

```
Begin
    Set NP=10; F=0.8; CR=0.8;MSCR=0.95
    Input α1, α2
    Initialize a random population
    For t=1 to MGN
        For j=1 to NP
            Randomly generate three integers in the range [1,NP], and then determine mutant vector
            If rand(0,1)<MSCR
                m^{j,t} = y^{1,t} + F(y^{2,t} − y^{3,t})
            Else
                m^{j,t} = y^{1,t} + F(y^{best,t−1} - y^{2,t})
            End If
            Randomly generate an integer i_rand within the range [1,N]
            For i=1 to N
                If rand(0,1)<CR or i = i_rand
                    r_i^{j,t} = m_i^{j,t}
                Else
                    r_i^{j,t} = y_i^{j,t}
                End If
            End For
            Determine the objective function values of trial and target vectors (r^t,y^t)
            If f(r^t) < f(y^t)
                y^{t+1} = r^t
            Else
                y^{t+1} = y^t
            End If
        End For
        Local search started
        Find best solution vector in the current population at generation t
        Generate dx random vector within [α1,α2] range
        y^{new,t} = y^{best,t} + dx
        Determine the objective function value of new target vector
            If f(y^{new,t}) < f(y^{best,t})
                y^{best,t} = y^{new,t}
            Else
                Determine new value of target vector in other direction
                y^{new,t} = y^{best,t} − dx
                If f(y^{new,t}) < f(y^{best,t})
                    y^{best,t} = y^{new,t}
                End If
            End If
            Decrease the length of jump for next generation
            dx = dx * 0.9
        Local search ended
        Stopping criterion
            If |Z_best − Z_average| / |Z_best| ≤ 10^−3 , then quit. Otherwise, go to next generation.
    End For
End
```

Fig. 1. Pseudo-code of MODE

5. Numerical application

In order to show the performance of the MODE algorithm against the standard DE in solving the CNDP, a small-sized example test network, which is adopted from Xu et al. (2009), is used. This network consists of 18 links and 6 nodes as can be seen in Fig. 2. The travel demand scenarios for this network are given in Table 1.

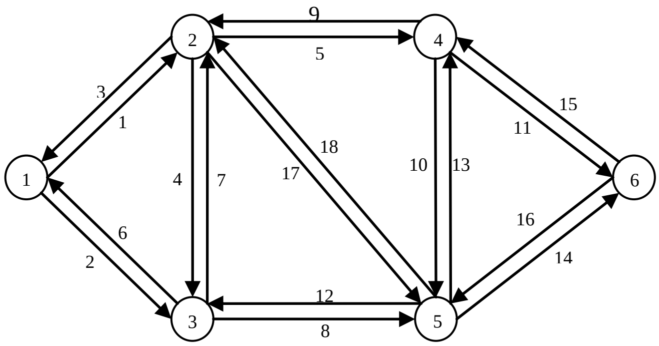

Fig. 2. 18-link network

Table 1. Travel demands

Scenario	D_{16}	D_{61}	Total demand
1	5	10	15
2	15	25	40

The travel time function is defined as shown in Eq. (11), and its corresponding parameters for 18-link network can be taken from Xu et al. (2009).

$$t_a(x_a, y_a) = \alpha_a + \beta_a \left(\frac{x_a}{\theta_a + y_a}\right)^4 \tag{11}$$

The upper level objective function is defined as:

$$\min_{\mathbf{x},\mathbf{y}} Z(\mathbf{x},\mathbf{y}) = \sum_{a \in \boldsymbol{A}} (t_a(x_a, y_a) x_a + d_a\, y_a) \tag{12}$$

$$\text{s.t.} \qquad 0 \le y_a \le u_a \,, \qquad \forall a \in A \tag{13}$$

Upper bound for capacity expansion of link $a \in A$ is set to 20 for fair comparison with results generated by Xu et al. (2009). The comparison of CPU times is conducted in terms of the number of UE assignments (NUE) since traffic assignment process is the most time consuming part of the algorithms. Results obtained through the DE and MODE are also compared with those produced by the SA and GA algorithms on the same network taken from Xu et al. (2009), and they are given in Table 2. As it can be seen from Table, the MODE shows steady convergence towards the global or near-global optimum for all demand scenarios and achieved good solutions in terms of the objective function value and the number of UE assignments. In scenario 1, the DE algorithm converges to the value of 190.43 after 86 generations (i.e. 860 number of UE assignments) while the MODE achieves slightly better objective function value of 190.33 with much less number of UE assignments than the DE. In addition, the SA and GA algorithms produced the value of 205.89 and 191.26 after 15000 and 50000 number of UE assignments, respectively (Xu et al., 2009). It is clear that the DE and MODE algorithms produce much better results than the SA and GA in terms of the objective function value and number of UE assignments.

In comparison with the results generated by the DE, SA and GA algorithms, it has been found that the MODE is capable of solving the CNDP with much less computational efforts.

Table 2. Results on the 18-link network

	Scenario 1				Scenario 2			
	MODE	DE	SA	GA	MODE	DE	SA	GA
y_1	0	0	0	0	0	0	0	0
y_2	0	0	0.47	0	10.13	9.29	9.12	11.98
y_3	0	0	0.65	0	17.54	17.34	18.12	16.24
y_4	0	0	0	0	0	0	0	0
y_5	0	0	0	0	0	0	0	0
y_6	5.24	4.93	6.53	4.47	5.95	6.12	4.98	5.40
y_7	0	0	0.80	0	0	0	0.11	0
y_8	0	0	0.25	0	3.48	2.69	1.58	6.04
y_9	0	0	0	0	0	0.12	0	0
y_{10}	0	0	0	0	0.01	0.04	0	0
y_{11}	0	0	0	0	0	0	0	0
y_{12}	0	0	0	0	0	0	0	0
y_{13}	0	0	0	0	0	0	0	0
y_{14}	0	0	0.84	0	12.66	12.25	11.66	12.28
y_{15}	0	0	0.14	0	0	0	2.97	0.82
y_{16}	7.61	7.69	7.34	7.54	19.99	19.94	19.71	19.99
y_{17}	0	0	0	0	0	0.12	0	0
y_{18}	0	0	0	0	0	0	0	0
Z	***190.33***	190.43	205.89	191.26	***729.58***	730.43	739.54	744.39
NUE	***396***	860	15000	50000	***759***	1090	22500	50000

For scenario 1, the convergence trend of the MODE and DE algorithms is given in Fig. 3. As can be seen in Fig. 3, the MODE algorithm with developed mutation strategy and local search operator converges rapidly without being trapped in bad local optimum after almost 20 generations. The reason how the MODE converges so rapidly to the near global optimum is that it uses information of the best solution vector produced at the previous generation with new mutation strategy. The other one, it takes advantage of the local search operator after each generation in which the best solution may be pushed to be located closer to the near global optimum solutions.

To investigate the ability of the MODE in solving the CNDP under heavy demand condition, scenario 2 has been considered and results are obtained as given in Fig. 4. In scenario 2, the DE reached the near global optimum value of 730.43 after about 1100 UE assignments as shown in Fig. 4. Although the MODE slightly outperformed than the DE for given stopping criterion, the MODE reaches the optimal solution with much less UE assignments in comparison with the DE. Besides, it is remarkable that the results produced by the SA and GA are not as good as those generated by the DE and MODE in terms of both objective function value and required number of UE assignments as can be seen in Table 2.

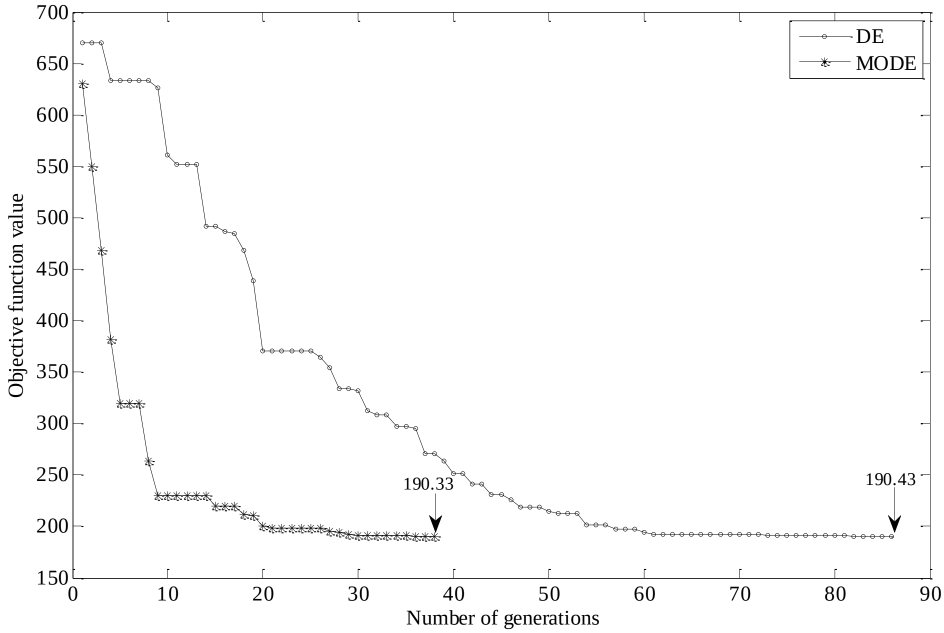

Fig. 3. Convergence comparisons of the DE and MODE for scenario 1

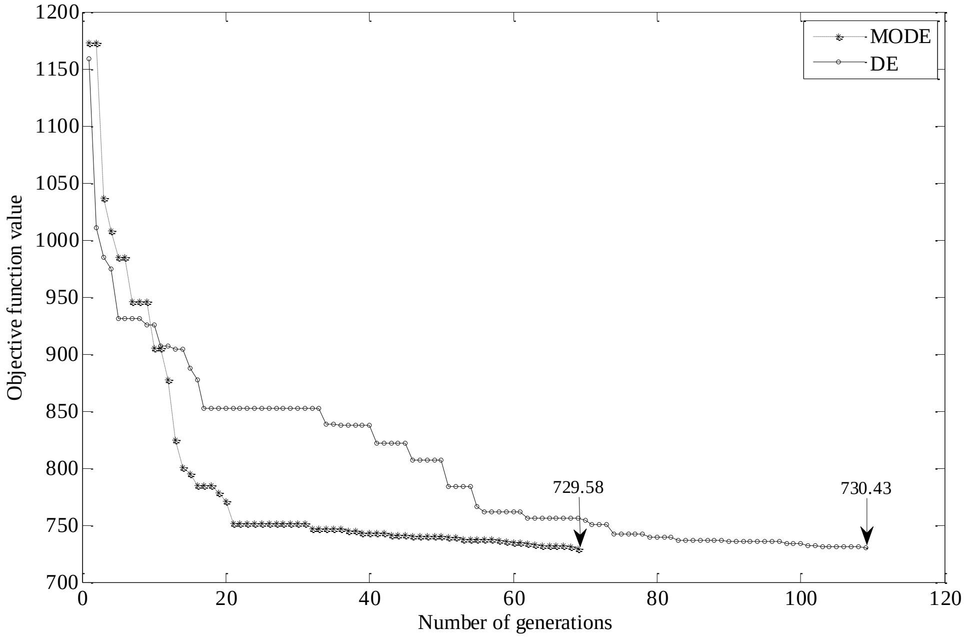

Fig. 4. Convergence comparisons of the DE and MODE for scenario 2

According to the results obtained from numerical experiments, it has been found that the MODE is much more efficient and effective method than the DE, SA and GA in terms of the objective function value and required number of UE assignments in solving the CNDP.

6. Conclusions

In this paper, an efficient heuristic algorithm called MODE was used to solve the CNDP with link capacity expansions. The CNDP was modeled as a bilevel programming model which is nonconvex. The upper level objective function has been defined as sum of the total travel time and total investment costs of link capacity expansions on the network while the lower level problem is formulated as user equilibrium traffic assignment model. The Frank-Wolfe method was used to solve the traffic assignment problem at the lower level. Numerical computations have been conducted on 18-link network and two demand scenarios were proposed. According to the results, the MODE achieved better solutions in all scenarios in terms of objective function value and number of UE assignments in comparison with the DE. Moreover, the newly developed mutation strategy and local search operator have facilitated the rate of convergence of the base DE algorithm without being trapped in bad local optimum.

References

Abdulaal, M., & LeBlanc, L. (1979). Continuous equilibrium network design models. *Transportation Research Part B*, *13*, 19-32.

Ban, X. G., Liu, H. X., Lu, J. G., & Ferris, M.C. (2006). Decomposition scheme for continuous network design problem with asymmetric user equilibria. *Transportation Research Record (1964)*, 185-192.

Chiou, S. W. (2005). Bilevel programming for the continuous transport network design problem. *Transportation Research Part B, 39*, 361-383.

Cho, H. J. (1988). *Sensitivity analysis of equilibrium network flows and its application to the development of solution methods for equilibrium network design problems*. PhD Dissertation, University of Pennsylvania, Philadelphia.

Davis, G. A. (1994). Exact local solution of the continuous network design problem via stochastic user equilibrium assignment. *Transportation Research Part B*, *28*(*1*), pp. 61-75.

Friesz, T. L., Cho, H. J., Mehta, N. J., Tobin, R. L., & Anandalingam, G. (1992). A simulated annealing approach to the network design problem with variational inequality constraints. *Transportation Science*, 18-26.

Friesz, T. L., Tobin, R. L., Cho, H. J., & Mehta, N. J. (1990). Sensitivity analysis based algorithms for mathematical programs with variational inequality constraints. *Mathematical Programming, 48*, 265-284.

Karoonsoontawong, A. & Waller, S. T. (2006). Dynamic continuous network design problem-Linear bilevel programming and metaheuristic approaches. *Transportation Research Record (1964)*, 104-117.

Li, C., Yang, H., Zhu, D., & Meng, Q. (2012). A global optimization method for continuous network design problems. *Transportation Research Part B, 46*, 1144-1158.

Liu, H., Cai, Z., & Wang, Y. (2010). Hybridizing particle swarm optimization with differential evolution for constrained numerical and engineering optimization. *Applied Soft Computing*, 629-640.

Marcotte, P., & Marquis, G. (1992). Efficient implementation of heuristics for the continuous network design problem. *Annals of Operational Research, 34*, 163-176.

Meng, Q., Yang, H., & Bell, M. G. H. (2001). An equivalent continuously differentiable model and a locally convergent algorithm for the continuous network design problem. *Transportation Research Part B, 35*, 83-105.

Storn, R., & Price, K. (1995). *Differential evolution: A simple and efficient adaptive scheme for global optimization over continuous spaces*. ICSI, USA, Tech. Rep. TR-95-012.

Suwansirikul, C., Friesz, T. L., & Tobin, R. L. (1987). Equilibrium decomposed optimisation: a heuristic for the continuous equilibrium network design problem. *Transportation Science*, 21(4), 254-263.

Tan, H. N., Gershwin, S. B., & Athans, M. (1979). *Hybrid Optimization in Urban Traffic Networks*. Report No. DOT-TSCRSPA-79-7, Laboratory for Information and Decision System. MIT Press, Cambridge, MA.

Wang, D. Z. W., & Lo, H. K. (2010). Global optimum of the linearized network design problem with equilibrium flows. *Transportation Research Part B, 44(4)*, 482–492.

Xu, T., Wei, H., & Hu, G. (2009). Study on continuous network design problem using simulated annealing and genetic algorithm. *Expert Systems with Applications*, *36*, 1322-1328.

Yang, H. (1995). Sensitivity analysis for queuing equilibrium network flow and its application to traffic control. *Mathematical and Computer Modelling, 22(4-7)*, 247–258.

Yang, H. (1997). Sensitivity analysis for the network equilibrium problem with elastic demand. *Transportation Research, 31*, 55-70.