Hindawi Publishing Corporation
Journal of Applied Mathematics
Volume 2013, Article ID 718015, 11 pages
http://dx.doi.org/10.1155/2013/718015

Research Article

Determining Optimal Link Capacity Expansions in Road Networks Using Cuckoo Search Algorithm with Lévy Flights

Ozgur Baskan

Department of Civil Engineering, Faculty of Engineering, Pamukkale University, 20070 Denizli, Turkey

Correspondence should be addressed to Ozgur Baskan; obaskan@pau.edu.tr

Received 28 June 2013; Accepted 31 July 2013

Academic Editor: Xin-She Yang

During the last two decades, Continuous Network Design Problem (CNDP) has received much more attention because of increasing trend of traffic congestion in road networks. In the CNDP, the problem is to find optimal link capacity expansions by minimizing the sum of total travel time and investment cost of capacity expansions in a road network. Considering both increasing traffic congestion and limited budgets of local authorities, the CNDP deserves to receive more attention in order to use available budget economically and to mitigate traffic congestion. The CNDP can generally be formulated as bilevel programming model in which the upper level deals with finding optimal link capacity expansions, whereas at the lower level, User Equilibrium (UE) link flows are determined by Wardrop's first principle. In this paper, cuckoo search (CS) algorithm with Lévy flights is introduced for finding optimal link capacity expansions because of its recent successful applications in solving such complex problems. CS is applied to the 16-link and Sioux Falls networks and compared with available methods in the literature. Results show the potential of CS for finding optimal or near optimal link capacity expansions in a given road network.

1. Introduction

The Continuous Network Design Problem (CNDP) deals with finding optimal capacity expansions for a set of selected links and the corresponding equilibrium link flows in a given road network. This well-known transportation problem has been studied for many years to improve the performance of transportation road networks and thus mitigate traffic congestion. Since the multiple objectives exist in the CNDP, it is considerably typical to formulate it as a bilevel model, which is difficult to solve. Due to the nonconvexity of the bilevel solution of the CNDP, it can be recognized as one of the most challenging problems in transportation field. The difficulty of the bilevel modeling arises from the evaluation of the upper level objective function that involves solving the lower level problem for each set of upper level decision variables. In the CNDP, upper level problem can be formulated by minimizing the sum of total travel time and total investment cost of link capacity expansions in a given road network, whilst the lower level can be solved as User Equilibrium (UE) traffic assignment model considering Wardrop's first principle [1]. In the CNDP, optimal capacity expansion plan at the upper level cannot be found without considering the reactions of the road users to that plan at the lower level. Although the upper and lower level problems consist of convex problems, the bilevel solution of the CNDP may be nonconvex due to both traffic assignment constraints and nonlinear travel time function. This nonconvexity may cause serious problems for deterministic algorithms [2].

The first continuous network design formulation was proposed by Abdulaal and LeBlanc [3]. They have formulated the network design problem with continuous variables subject to equilibrium assignment as a nonlinear optimization problem. Hooke-Jeeves (HJ) and Powell's methods were used in order to test the proposed model on a medium-sized network. In their study, the effect of type of investment function was also investigated. It has been found that the performance of two methods is approximately same for convex investment cost function, whilst the HJ is better than the Powell's method in the case of using concave investment function. After this first study, several variations of the CNDP have been studied and various solution methods have been proposed for solving this problem. Suwansirikul et al. [4] proposed Equilibrium Decomposition Optimization (EDO) method for finding an approximate solution to the CNDP and tested this heuristic on several networks. Their results showed that the proposed heuristic is more efficient than the HJ algorithm. The efficiency of the method stems from decomposition of the original problem into a set of suboptimization problems. The other advantage of the EDO algorithm was reported that the computational cost of the proposed model does not depend on the number of links which are considered for capacity expansion. Marcotte [5] and Marcotte and Marquis [6] presented efficient implementations of heuristic methods in small-sized networks for solving the CNDP where road users follow Wardrop's first principle. In addition, a number of sensitivity-based heuristic algorithms were developed for the CNDP [7–10]. Furthermore, Friesz et al. [11] used Simulated Annealing (SA) approach to solve the CNDP for two different road networks and found that the proposed heuristic is more efficient than Iterative Optimization Assignment (IOA), HJ, and EDO algorithms. Afterwards, Friesz et al. [12] presented a model for continuous multiobjective optimal design of a transportation road network. Results showed that SA is ideally suited for solving multiobjective versions of the equilibrium network design problem. Unlike using classical lower level solution as in most studies, Stochastic User Equilibrium (SUE) assignment procedure was embedded to the CNDP in Davis [13]. The generalized reduced gradient and sequential quadratic programming methods were used to solve the CNDP. The proposed solution methods were tested on several example networks, and it has been found that the differentiable and tractable version of the CNDP could be created. In order to avoid the disadvantages for the use of bilevel formulation, Meng et al. [14] formulated the CNDP as a single level continuously differentiable optimization problem and applied the augmented Lagrangian method to solve this problem. Their results showed that the proposed method has the potential for application to large-scale problems. Chiou [15] used a bilevel programming model to solve the CNDP. Four variants of gradient-based methods are presented, and numerical comparisons are made with several test networks. Results showed that the proposed methods are more effective than the other compared algorithms when especially congested road networks are considered.

Similarly, Ban et al. [16] proposed a relaxation method to solve the CNDP when the lower level is a nonlinear complementary problem. They converted original bilevel model into a single level formulation by means of adding some constraints to the lower level problem, and a relaxation scheme was proposed to solve it. The proposed solution algorithm was tested on different test networks, and promising results were obtained. Karoonsoontawong and Waller [17] presented SA, Genetic Algorithm (GA), and random search techniques to solve the CNDP. Their study showed that GA is performed better than the others on the test networks in terms of solution quality and convergence. Moreover, they emphasized that the algorithm parameters should be calibrated to achieve best results for different road networks. Gao et al. [18] converted the bilevel solution of the CNDP into a single level convex model and proposed a globally convergent algorithm to solve this problem. They presented a numerical example to show the effectiveness of the proposed method against the other existing heuristic algorithms. Xu et al. [19] proposed SA and GA methods to achieve the optimal solutions of the CNDP. They tested the proposed methods on small-sized network for three demand scenarios and found that when demand is large, SA is more efficient than GA in solving the CNDP. In addition, much more computational time is needed for GA in comparison with SA in order to achieve same optimal solution. Unlike the study proposed by Xu et al. [19], Mathew and Sharma [20] reported that GA model is more efficient than other compared algorithms available in the literature for solving the CNDP. They applied the proposed model to three different road networks, one of which is considered as a real city network, and found that the GA is capable of finding high quality solution especially for large scale road network. Wang and Lo [21] formulated the CNDP as a single level optimization problem subject to equilibrium constraints. In order to overcome the nonconvexity of the CNDP, they transformed the equilibrium constraints into a set of mixed-integer constraints and linearized the travel time function. Their results showed that the proposed method is capable to find globally optimal solution of the CNDP. Li et al. [22] presented an applicable global optimization method for solving the CNDP and converted the CNDP into a sequence of single level concave programs. Their method has the potential to find global optimum of large network design problems. Baskan and Dell'Orco [23] applied artificial bee colony optimization algorithm to solve the CNDP. The proposed method is compared with SA and GA algorithms for small-sized road network and obtained good results in comparison with other methods in terms of objective function value and number of UE assignments. In addition, Baskan [2] and Baskan and Ceylan [24] attempted to solve the bilevel formulation of the CNDP using Harmony Search and Differential Evolution algorithms, respectively.

From the viewpoint of reserve capacity, Yang and Wang [25] compared the level of equivalence and effectiveness of two different objective functions for the CNDP, which are minimizing the total system cost under a budget constraint and maximization of network reserve capacity. Numerical results showed that a combined objective function by applying different weightings on two objectives can also be more effective. Following the study made by [25], Ziyou and Yifan [26] combined the concept of reserve capacity with the CNDP. A bilevel programming model and heuristic solution algorithm based on sensitivity analysis are proposed to solve the reserve capacity problem of optimal signal control with user equilibrium route choice. They concluded that it is crucial importance to combine the concept of reserve capacity with the CNDP in order to provide more realistic information for transportation planners. Similarly, Chiou [27] proposed a projected Quasi-Newton method for simultaneously solving the problem of finding the maximum possible increase in travel demand and determining optimal link capacity expansions. Numerical applications are made on signal-controlled networks where obtained results outperform than traditional methods.

The reviewed literature shows that heuristic methods play an important role for solving the various types of the CNDP. Therefore, the performance of recently developed heuristic algorithms needs to be investigated in order to obtain probably better solutions to the CNDP. On the other hand, although some exact algorithms for the CNDP are available in the literature, they may not be suitable especially for large scale networks. Therefore, this paper deals with finding optimal link capacity expansions in a given road network using cuckoo search (CS) algorithm with Lévy flights. For this purpose, a bilevel model has been proposed in which the lower level problem is formulated as UE traffic assignment mode,l and Frank-Wolfe (FW) method is used to solve this problem.

The remainder of this paper is organized as follows. In Section 2, problem formulation for the CNDP is given. In the next section, the CS algorithm and its solution procedure on the CNDP are presented. In Section 4, numerical studies are conducted on two different test networks. Finally, concluding remarks and future study directions are given in Section 5.

2. Problem Formulation

The following notation is used for the problem formulation:

A : the set of links, $\forall a \in A$,

C_{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,

D: the vector of O-D pair demands, $D = [D_{rs}]$ $\forall r \in R$, $s \in S$,

f: the vector of path flows, $f = [f_c^{rs}]$, $\forall r \in R$, $s \in S$, $c \in C_{rs}$,

g: the vector of investment costs, $g = [g_a(y_a)]$ $\forall a \in A$,

L: the step length,

t: the vector of link travel times, $t = [t_a(x_a, y_a)]$ $\forall a \in A$

h: the vector of upper bound for link capacity expansions, $h = [h_a]$, $\forall a \in A$,

x: the vector of equilibrium link flows, $x = [x_a]$, $\forall a \in A$,

y: the vector of link capacity expansions, $y = [y_a]$, $\forall a \in A$,

$\mathbf{P}$: the probability matrix,

$\mathbf{K}$: the matrix of local step size,

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

n: the number of nests,

p: the discovering parameter,

Z: upper level objective function,

z: lower level objective function,

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

ρ: the conversion factor from investment cost to travel times,

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

λ_a, φ_a: the parameters of link cost function, $\forall a \in A$,

u, v: the parameters of the step length L,

β: the scale parameter,

Γ: the gamma function,

α: the step size.

2.1. Upper Level Formulation. In the CNDP, the upper level deals with finding optimal capacity expansion plan for a set of selected links in a given road network by minimizing the total system cost and construction cost, while the lower level determines UE link flows considering given capacity expansion plan in the upper level. Therefore, the CNDP is recognized within the framework of a leader-follower, where the transportation planner is the leader and the user is the follower [28]. It is assumed that the leader as transportation planning manager can influence the users' route choice behavior but cannot be able to control it. The users make their route choice decision by minimizing their own travel costs under given service level of transportation road networks [18]. This interaction can be formulated as follows:

$$\begin{aligned} \min_{y} \quad & Z(x, y) = \sum_{a \in A} \left(t_a(x_a, y_a)\, x_a + \rho g_a(y_a)\right) \\ \text{s.t.} \quad & 0 \le y_a \le h_a, \quad \forall a \in A \\ & x = x(y), \end{aligned} \tag{1}$$

where $x(y)$ is UE link flow under given capacity expansion plan and obtained by solving the lower level problem. The constraint of (1) ensures that the investment cost of link $a\, \forall a \in A$ will not exceed the related budget. It is also the nonnegativity constraint of the upper level decision variables.

2.2. Lower Level Formulation. In the CNDP models, the user's route choice behavior is generally characterized by the UE assignment that is to find the equilibrium link flows. In this paper, Wardrop's first principle is followed, which states that the travel times of all used paths between the same Origin-Destination (O-D) pair are equal and less than any unused paths [21]. It is well-known that the increase in capacity of any link in a given road network without considering the response of network users may actually increase traffic congestion rather than reducing it. Due to well-known Braess' paradox, prediction of traffic flows is crucial importance to the CNDP. The UE assignment problem can be formulated as follows:

$$\begin{aligned} \min_{x} \quad & z = \sum_{a \in A} \int_0^{x_a} t_a(w, y_a)\, dw \\ \text{s.t.} \quad & \sum_{c \in C} f_c^{rs} = D_{rs}, \quad \forall r \in R,\ s \in S,\ c \in C_{rs} \\ & x_a = \sum_{rs} \sum_{c \in C_{rs}} f_c^{rs} \delta_{a,c}^{rs}, \quad \forall r \in R,\ s \in S,\ a \in A,\ c \in C_{rs} \\ & f_c^{rs} \ge 0, \quad \forall r \in R,\ s \in S,\ c \in C_{rs}, \end{aligned} \tag{2}$$

where the constraints of (2) are definitional, conservation of the flow constraints, and nonnegativity, respectively. Since UE traffic assignment is a convex optimization problem, it can be numerically solved by various methods. For this purpose, the most widely used solution method is the Frank-Wolfe (FW) algorithm [29]. It has been developed for solving quadratic optimization problems and is also highly convenient for determining equilibrium link flows in road networks [30].

3. Cuckoo Search Algorithm

3.1. Cuckoo Breeding Behavior. The CS is an optimization algorithm proposed by Yang and Deb [31, 32] and recently improved for multiobjective optimization [33]. Before the description of the CS algorithm, it may be helpful to briefly review the fascinating breed behavior of some cuckoo species. The CS is inspired by some cuckoo species by laying their eggs in the nests of host birds of other species. In this case, if a host bird realizes that the eggs are not their own, these alien eggs are either taken away or the nest is abandoned by host bird, and a new nest is built elsewhere. Some cuckoo species such as parasitic cuckoos have evolved in such a way that female cuckoos are very specialized in the mimicry in colours and pattern of the eggs of a few chosen host species. This behavior reduces the probability of their eggs being abandoned and thus increases their hatching probability [34]. Additionally, a cuckoo chick can imitate the call of host chicks to gain access to more feeding opportunities [35]. Furthermore, the timing of egg laying of some cuckoo species is also amazing. Parasitic cuckoos often choose a nest where the host bird has just laid its own eggs. In general, the cuckoo eggs hatch slightly earlier than the host eggs. Once the first cuckoo chick is hatched, the first instinctive action by cuckoo chicks is to evict the host eggs by blindly propelling the eggs out of the nest, which increases the cuckoo chick's share of food provided by the host bird [34]. As summarized previously, the CS algorithm idealizes cuckoo's breeding behavior and thus can be applied to various optimization problems.

3.2. Lévy Flights. As it is well-known, random searching is crucial importance in meta-heuristic algorithms. The Lévy flight is a random process which consists of taking a series of consecutive random steps [36]. From the mathematical point of view, two consecutive steps need to be performed to generate random numbers with Lévy flights: (i) the generation of steps and (ii) the choice of a random direction. To do this, one of the most efficient methods is to use the so-called Mantegna algorithm where the step length L can be determined as follows:

$$L = \frac{u}{|v|^{1/\beta}}, \qquad (3)$$

where β is the scale parameter and its recommended range is $[1, 2]$. The β value is set to 1.5 in this study. u and v are obtained from normal distribution as

$$u \sim N\left(0, \sigma_u^2\right), \qquad v \sim N\left(0, \sigma_v^2\right), \qquad (4)$$

where σ_u and σ_v are calculated using the following:

$$\sigma_u = \left\{ \frac{\Gamma(1+\beta)\sin(\pi\beta/2)}{\Gamma[(1+\beta)/2]\,\beta 2^{(\beta-1)/2}} \right\}^{1/\beta}, \qquad \sigma_v = 1, \qquad (5)$$

where Γ denotes gamma function.

3.3. Cuckoo Search. The CS algorithm is inspired by some species of cuckoos because of their special lifestyle and fascinating breeding behavior [37]. These species tend to lay their eggs in the nests which belong to other host birds. Regarding this parasite behavior of some species of cuckoos, some of host birds either throw out these alien eggs or abandon their nests and build new nests elsewhere.

The following three rules are utilized in the CS algorithm: (i) selection of the best by keeping the best nests or solutions (ii) replacement of host eggs with respect to the quality of the new solutions or cuckoo eggs produced based randomization with Lévy flights and (iii) discovering of some cuckoo eggs by the host birds and replacing according to the quality of the local random walks [38]. The algorithm steps of CS are based on these rules and given in Algorithm 1.

One of the most important steps in the algorithm is the use of Lévy flights for random searching. The Lévy flight is a type of random walk and described by a series of instantaneous jumps chosen from a probability density function which has a power law tail [39, 40]. The step size α, which controlled random search process in Lévy flight, is generally selected between $[0, 1]$ interval. Setting $\alpha = 0.1$ may produce efficient results especially for small-sized optimization problems [40]. The other important parameter in the CS algorithm is p which is used by discovering of cuckoo eggs by the host birds. Besides Yang and Deb [32] emphasized that the convergence rate of the algorithm was not strongly affected by the p value; they suggested setting $p = 0.25$. The flowchart of the CS algorithm for the CNDP is given in Figure 1 and corresponding solution steps can be summarized as given in Figure 1.

Initialize the CS Parameters. The CS parameters, which are number of nests (n), step size (α), discovering probability (p), and maximum number of generations (MNG), are set to 10, 0.1, 0.25, and 1000, respectively. These values are selected in accordance with the recommendation by Yang and Deb [32].

Generation of Initial Population. At generation t, the nests, $\text{nest}^t = [\text{nest}_i^t]$, where $i \in \{1, 2, \ldots, n\}$, are filled with randomly generated capacity expansions for a set of selected links in a given road network by considering upper and lower bounds, and UE link flows are determined for each nest (i.e., set of link capacity expansions) by using (2). After that, their corresponding objective function values are calculated using (1).

Generate New Nests by Lévy Flights. The vector of new nest is generated from randomly selected ith nest by Lévy flights using the following:

$$\text{new\_nest}_i^t = \text{nest}_i^t + \alpha L\left(\text{nest}_i^t - \text{nest}_{\text{best}}^t\right), \qquad (6)$$

Generate initial population of n nests $y_i, i = 1, 2, \ldots, n$
for $i = 1 : n$
  Calculate fitness value $F_i = f(y_i)$
end for
while (stopping criterion is not satisfied)
  Generate a cuckoo egg (y_j) by Lévy flights from random nest
  Calculatefitness value $F_j = f(y_j)$
  Choose a random nest i
   if $F_j > F_i$ **then**
   $y_i \leftarrow y_j$
   $F_i \leftarrow F_j$
   end if
  Abandon a fraction p_a of the worst nests
  Build new nests by Lévy flights
  Evaluate fitness of new nests and keep best nest
end while

Algorithm 1: Cuckoo search algorithm.

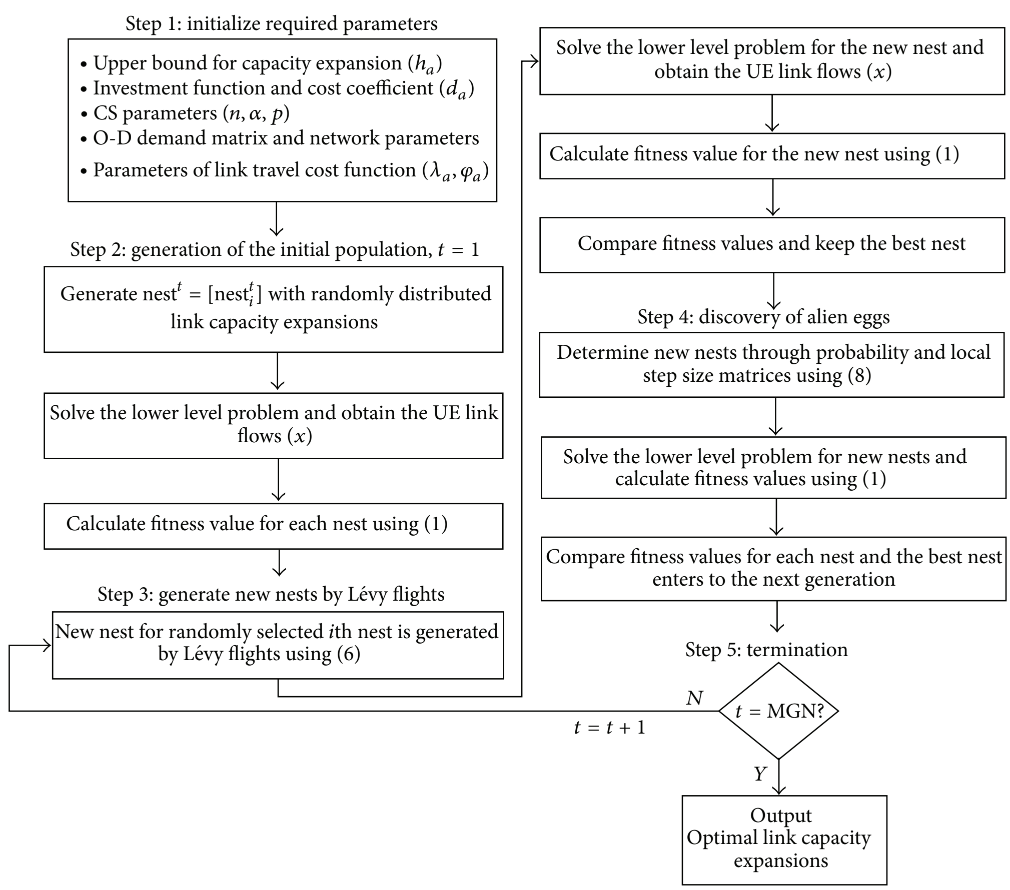

Figure 1: Flowchart of the CS for the CNDP.

where new_nest$_i^t$ is the new nest generated by Lévy flights; nest$_i^t$ is randomly selected nest from population; nest$_{\text{best}}^t$ is the best nest obtained so far; α is step size; and L is the Lévy flights vector or step length as in Mantegna's algorithm. After determining the new nest, the objective function values of two nests are calculated using (1), and the best nest is kept.

Discovery of Alien Eggs. The alien eggs discovery is performed for all of the eggs using the probability matrix. The probability matrix is produced as

$$p_{ij} = \begin{cases} 1, & \text{if } \operatorname{rand}(0,1) < p \\ 0, & \text{otherwise}, \end{cases} \quad (7)$$

where p_{ij} is discovering probability for the jth variable of the ith nest. The value of p is compared with the output of a uniform random number generator, rand(0, 1), to determine whether local random walk is considered or not. After determining discovering probabilities, new nests are produced using the following:

$$\text{new\_nest}^t = \text{nest}^t + \mathbf{K} * \mathbf{P}, \quad (8)$$

where $\mathbf{P}$ is the probability matrix and $\mathbf{K}$ is the matrix of local step size, which is produced by using the following:

$$\mathbf{K} = \text{rand}\,() * (\text{nests}\,[\text{permute}\,1\,[i]]\,[j] - \text{nests}\,[\text{permute}\,2\,[i]]\,[j]), \quad (9)$$

where rand( ) is random number generator in [0, 1] interval and permute 1 and permute 2 are different rows permutation functions applied to the nests matrix [41]. Finally, the existing and new objective function values are compared for each nest and the best nest enters to the next generation according to the simple rule as given in the following:

$$\text{nest}_i^{t+1} = \begin{cases} \text{nest}_i^t, & \text{if } F(\text{nest}_i^t) < F(\text{new\_nest}_i^t) \\ \text{new\_nest}_i^t, & \text{otherwise.} \end{cases} \quad (10)$$

Termination. The generation of new nests and the discovering of the alien eggs steps are repeated until a predetermined stopping criterion is satisfied or maximum number of generations is reached.

4. Numerical Studies

4.1. 16-Link Network. In order to test the performance of the CS algorithm in solving the CNDP, it is first applied to the 16-link network which is most widely used test network by many researchers. This network consists of 16 links and 6 nodes as shown in Figure 2. For this network, two demand scenarios are considered as given in Table 1. Results obtained by the CS algorithm for different demand cases are compared with the results produced by other methods available in the literature. The compared methods for all numerical applications are given in Table 2. The link travel time function is defined as given in (11). Link parameters and demand data are adopted from Suwansirikul et al. [4]. Consider the following:

$$t_a(x_a, y_a) = \lambda_a + \varphi_a\left(\frac{x_a}{\theta_a + y_a}\right)^4. \quad (11)$$

The upper level objective function for the 16-link network is defined as

$$\min_y \quad Z(x, y) = \sum_{a \in A} (t_a(x_a, y_a)\, x_a + d_a y_a)$$
$$\text{s.t.} \quad 0 \le y_a \le h_a, \quad \forall a \in A, \quad (12)$$

where d_a is the cost coefficient. The upper bound (h_a) was set to 10 and 20 for scenarios 1-2 for fair comparison with other

Table 1: Travel demand scenarios for the 16-link network.

Scenario	D_{16}	D_{61}	Total demand
1	5	10	15
2	10	20	30

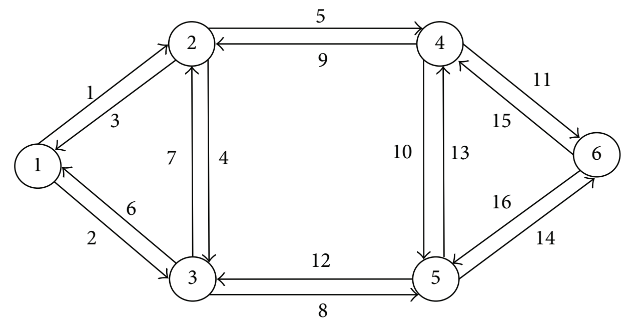

Figure 2: 16-link network.

algorithms. Results obtained by all algorithms for scenario 1 are presented in Tables 3 and 4.

Since the CS algorithm is a stochastic search method, the results obtained from this algorithm are selected as the best output of trials made by different random numbers. As can be seen from Table 4, the CS algorithm achieved to the value of 199.32 as its best output. Among all the compared algorithms, the SA produces the best solution but needs much more computational efforts than other algorithms in solving traffic assignment problem. It is clear that the CS produces better results with less computational efforts in comparison with other methods except SA, CG, and QNEW for scenario 1. However, they need much computational efforts than the CS algorithm in solving traffic assignment problem. Additionally, results show that the objective function values produced by all compared algorithms are very close to each other, but optimal capacity expansion vectors are not similarly consistent. This result shows us again that the CNDP has multiple optimal solutions.

In order to evaluate the sensitivity of the CS algorithm under different demand levels, scenario 2 is considered, and results are given in Tables 5 and 6. It can be clearly seen that the CS algorithm is able to produce the best solution among 14 algorithms, as well as with significant less computational efforts. In order to validate the obtained results for scenarios 1 and 2, the equilibrium link flows and travel times are given in Table 7.

4.2. Sioux Falls Network. In order to show the ability of the CS algorithm on realistic test network, the city of Sioux Falls is used which is probably the most used network for the CNDP. It consists of 24 nodes and 76 links. The link parameters of the network and travel demands between 552 O-D pairs are adopted from Suwansirikul et al. [4]. The link travel time function is used as given in (11). The dashed links 16, 17, 19, 20, 25, 26, 29, 39, 48, and 74 of Sioux Falls network are candidates for capacity expansion as shown in Figure 3. The upper level

Table 2: The compared algorithms on all test networks.

Methods	Sources
Hooke-Jeeves algorithm (HJ)	Abdulaal and LeBlanc [3]
Equilibrium Decomposed Optimization (EDO)	Suwansirikul et al. [4]
Modular In-core Nonlinear Optimization System (MINOS)	Suwansirikul et al. [4]
Genetic Algorithm (GA)	Mathew and Sharma [20]
Iterative Optimization Assignment algorithm (IOA)	Allsop [42]
Simulated Annealing algorithm (SA)	Friesz et al. [11]
Sensitivity Analysis Based algorithm (SAB)	Yang and Yagar [43]
Augmented Lagrangian algorithm (AL)	Meng et al. [14]
Path based Mixed Integer Linear Program (MILP)	Wang and Lo [21]
Link based Mixed Integer Linear Program (LMILP)	Luathep et al. [44]
Penalty with MultiCutting plane method (PMC)	Li et al. [22]
Gradient projection method (GP)	Chiou [15]
Conjugate gradient projection method (CG)	Chiou [15]
Quasi-Newton projection method (QNEW)	Chiou [15]
PARATAN version of gradient projection method (PT)	Chiou [15]
Cuckoo search (CS) algorithm with Lévy flights	This paper

Table 3: Comparison of results from solving the 16-link network for scenario 1.

	MINOS	HJ	EDO	IOA	SA	AL
y_1	0	0	0	0	0	0
y_2	0	0	0	0	0	0
y_3	0	1.2	0.13	0	0	0.0062
y_4	0	0	0	0	0	0
y_5	0	0	0	0	0	0
y_6	6.58	3.00	6.26	6.95	3.1639	5.2631
y_7	0	0	0	0	0	0.0032
y_8	0	0	0	0	0	0
y_9	0	0	0	0	0	0
y_{10}	0	0	0	0	0	0
y_{11}	0	0	0	0	0	0.0064
y_{12}	0	0	0	0	0	0
y_{13}	0	0	0	0	0	0
y_{14}	0	0	0	0	0	0
y_{15}	7.01	3.00	0.13	5.66	0	0.71701
y_{16}	0.22	2.80	6.26	1.79	6.7240	6.7561
⋮	⋮	⋮	⋮	⋮	⋮	⋮
Z	211.25	215.08	201.84	210.86	198.10378	202.9913
⋮	⋮	⋮	⋮	⋮	⋮	⋮
#	—	54	10	9	18300	2700

Note: Z describes the objective function value, and # denotes the number of Frank-Wolfe iterations performed.

Table 4: Comparison of results from solving the 16-link network for scenario 1 (continued).

	SAB	GP	CG	QNEW	MILP	CS
y_1	0	0	0	0	0	0
y_2	0	0	0	0	0	0
y_3	0	0	0	0	0	0
y_4	0	0	0	0	0	0
y_5	0	0	0	0	0	0
y_6	5.8352	5.8302	6.1989	6.0021	4.41	5.1894
y_7	0	0	0	0	0	0
y_8	0	0	0	0	0	0
y_9	0	0	0	0	0	0
y_{10}	0	0	0	0	0	0
y_{11}	0	0	0	0	0	0
y_{12}	0	0	0	0	0	0
y_{13}	0	0	0	0	0	0
y_{14}	0	0	0	0	0	0
y_{15}	0.9739	0.87	0.0849	0.1846	0	0
y_{16}	6.1762	6.1090	7.5888	7.5438	7.70	7.6076
⋮	⋮	⋮	⋮	⋮	⋮	⋮
Z	204.7	202.24	199.27	198.68	199.781	199.32
⋮	⋮	⋮	⋮	⋮	⋮	⋮
#	6	14	7	12	—	3

Note: Z describes the objective function value, and # denotes the number of Frank-Wolfe iterations performed.

objective function for the Sioux Falls network is formulated as in the following:

$$\begin{aligned} \min_{y} \quad & Z(x, y) = \sum_{a \in A} \left(t_a(x_a, y_a) x_a + 0.001 d_a y_a^2 \right) \\ \text{s.t} \quad & 0 \le y_a \le h_a, \quad \forall a \in A. \end{aligned} \tag{13}$$

The results obtained by the CS algorithm on the Sioux Falls network are compared with those generated by other existing methods, and they are given in Tables 8 and 9. From these tables, it can be observed that the CS algorithm is able to produce the best solution among the compared major algorithms except SA. Although the SA slightly outperformed

Table 5: Comparison of results from solving the 16-link network for scenario 2.

	MINOS	HJ	EDO	IOA	SA	AL	SAB
y_1	0	0	0	0	0	0	0.0189
y_2	4.61	5.40	4.88	4.55	0	4.6153	2.2246
y_3	9.86	8.18	8.59	10.65	10.1740	9.8804	9.3394
y_4	0	0	0	0	0	0	0
y_5	0	0	0	0	0	0	0
y_6	7.71	8.10	7.48	6.43	5.7769	7.5995	9.0466
y_7	0	0	0.26	0	0	0.0016	0
y_8	0.59	0.90	0.85	0.59	0	0.6001	0.0175
y_9	0	0	0	0	0	0.001	0
y_{10}	0	0	0	0	0	0	0
y_{11}	0	0	0	0	0	0	0
y_{12}	0	0	0	0	0	0.1130	0.0816
y_{13}	0	0	0	0	0	0	0
y_{14}	1.32	3.90	1.54	1.32	0	1.3184	0.0198
y_{15}	19.14	8.10	0.26	19.36	0	2.7265	2.1429
y_{16}	0.85	8.40	12.52	0.78	17.2786	17.5774	18.9835
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
Z	557.14	557.22	540.74	556.61	528.497	532.71	536.084
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
#	—	134	12	13	24300	4000	45

Note: Z describes the objective function value, and # denotes the number of Frank-Wolfe iterations performed.

Table 6: Comparison of results from solving the 16-link network for scenario 2 (continued).

	GP	CG	QNEW	MILP	LMILP	PMC	CS
y_1	0.1013	0.1022	0.0916	0	0	0	0
y_2	2.1818	2.1796	2.1521	4.41	2.722	4.6905	4.6144
y_3	9.3423	9.3425	9.1408	10.00	9.246	9.9778	9.9419
y_4	0	0	0	0	0	0	0
y_5	0	0	0	0	0	0	0
y_6	9.0443	9.0441	8.8503	7.42	8.538	7.5554	7.3821
y_7	0	0	0	0	0	0	0
y_8	0.008	0.0074	0.0114	0.54	0	0.6333	0.5922
y_9	0	0	0	0	0	0	0
y_{10}	0	0	0	0	0	0	0
y_{11}	0	0	0	0	0	0	0
y_{12}	0.0375	0.0358	0.0377	0	0	0	0
y_{13}	0	0	0	0	0	0	0
y_{14}	0.0089	0.0083	0.0129	1.18	0	1.7664	1.3152
y_{15}	1.9433	1.9483	1.9706	0	0	0	0
y_{16}	18.9859	18.986	18.575	19.50	20.000	19.6737	20
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
Z	534.017	534.109	534.08	523.627	526.488	522.748	522.396
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
#	31	16	11	—	—	—	4

Note: Z describes the objective function value, and # denotes the number of Frank-Wolfe iterations performed.

than CS, the objective function values obtained by both algorithms are quite close. In addition, the CS algorithm produced good results with much less computational efforts in solving the traffic assignment problem in comparison with SA. It should be noted that AL, HJ, and GA algorithms have the potential to produce good results for solving the CNDP, but they require much more computational efforts in solving traffic assignment problem than CS.

As presented in the previous numerical application, the equilibrium link flows and travel times produced by the CS on the Sioux Falls network are given in Tables 10 and 11 in order to give an opportunity to the readers for validating the obtained result by the CS algorithm.

Table 7: The equilibrium link flows and travel times from solving 16-link network.

Scenario 1				Scenario 2			
t_1	1	x_1	0	t_1	1	x_1	0
t_2	2.3125	x_2	5	t_2	3.0961	x_2	10
t_3	3.5915	x_3	6.0287	t_3	4.291	x_3	15.3514
t_4	4	x_4	0	t_4	4	x_4	0
t_5	5	x_5	0	t_5	5	x_5	0
t_6	3.9215	x_6	3.9713	t_6	3.2153	x_6	4.6486
t_7	1	x_7	0	t_7	1	x_7	0
t_8	1.0625	x_8	5	t_8	1.7944	x_8	10
t_9	2.0025	x_9	6.0287	t_9	2.1081	x_9	15.3514
t_{10}	3	x_{10}	0	t_{10}	3	x_{10}	0
t_{11}	9	x_{11}	0	t_{11}	9	x_{11}	0
t_{12}	5.9805	x_{12}	3.9713	t_{12}	7.633	x_{12}	4.6486
t_{13}	4.0043	x_{13}	5.0101	t_{13}	4.2827	x_{13}	14.3194
t_{14}	2.1289	x_{14}	5	t_{14}	3.5986	x_{14}	10
t_{15}	9.0335	x_{15}	1.0186	t_{15}	9.8617	x_{15}	1.032
t_{16}	6.3125	x_{16}	8.9814	t_{16}	6.3622	x_{16}	18.968

5. Conclusions

In this paper, the CS algorithm with Lévy flights has been introduced to solve the CNDP, which is formulated as a bilevel programming model. In the upper level, the objective function is defined as the sum of total travel time and investment cost of link capacity expansions. The lower level is formulated as UE static traffic assignment problem, and Frank-Wolfe method is used to solve it.

The proposed model is first tested on the 16-link network, which is widely used network for solving the CNDP. Two scenarios are considered in order to evaluate the sensitivity of the CS algorithm to different demand levels. Results obtained by the proposed algorithm are compared with those generated by existing major methods in the literature. From the results, it has been found that the CS algorithm is able to produce good results for solving the CNDP, especially

Table 8: Comparison of results from solving the Sioux Falls network.

Initial value of y_a	HJ 1.0	EDO 12.5	SA 6.25	AL 12.5	IOA 12.5	SAB 12.5
y_{16}	3.8	4.59	5.38	5.5728	4.6875	5.7392
y_{17}	3.6	1.52	2.26	1.6343	3.9063	5.7182
y_{19}	3.8	5.45	5.50	5.6228	1.2695	4.9591
y_{20}	2.4	2.33	2.01	1.6443	1.6599	4.9612
y_{25}	2.8	1.27	2.64	3.1437	2.3331	5.5066
y_{26}	1.4	2.33	2.47	3.2837	2.3438	5.5199
y_{29}	3.2	0.41	4.54	7.6519	5.5651	5.8024
y_{39}	4.0	4.59	4.45	3.8035	4.6862	5.5902
y_{48}	4.0	2.71	4.21	7.3820	5.4688	5.8439
y_{74}	4.0	2.71	4.67	3.6935	6.2500	5.8662
⋮	⋮	⋮	⋮	⋮	⋮	⋮
Z	81.77	83.47	80.87	81.75	87.34	84.21
⋮	⋮	⋮	⋮	⋮	⋮	⋮
#	108	12	3900	2700	31	11

Note: the upper bound for y was set to 25 for CS. Z describes the objective function value, and # denotes the number of Frank-Wolfe iterations performed.

Table 9: Comparison of results from solving the Sioux Falls network (continued).

Initial value of y_a	GP 12.5	CG 12.5	QNEW 6.25	PT 12.5	GA —	CS —
y_{16}	4.8693	4.7691	4.9776	5.0237	5.17	5.0916
y_{17}	4.8941	4.8605	5.0287	5.2158	2.94	1.3515
y_{19}	1.8694	3.0706	1.9412	1.8298	4.72	6.4903
y_{20}	1.5279	2.6836	2.1617	1.5747	1.76	2.2995
y_{25}	2.7168	2.8397	2.6333	2.7947	2.39	2.9074
y_{26}	2.7102	2.9754	2.7923	2.6639	2.91	2.0515
y_{29}	6.2455	5.6823	5.7462	6.1879	2.92	3.6725
y_{39}	5.0335	4.2726	5.6519	4.9624	5.99	5.2202
y_{48}	3.7597	4.4026	4.5738	4.0674	3.63	3.4230
y_{74}	3.5665	5.5183	4.1747	3.9199	4.43	4.8798
⋮	⋮	⋮	⋮	⋮	⋮	⋮
Z	82.71	82.53	83.08	82.53	81.74	81.51
⋮	⋮	⋮	⋮	⋮	⋮	⋮
#	9	6	5	7	77	36

Note: the upper bound for y was set to 25 for CS. Z describes the objective function value, and # denotes the number of Frank-Wolfe iterations performed.

under heavier demand condition. Secondly, the performance of the CS algorithm is tested on the Sioux Falls city network. In comparison with the results obtained by the other major algorithms except SA, the CS algorithm achieved the best solution. Although the SA slightly outperformed than CS, it needs much more computational efforts in solving the traffic assignment problem. It is clear that the CS algorithm gives promising results in terms of objective function value and required computational effort and would be considered for large-scale road network applications in solving the CNDP.

Table 10: The resulting link travel times from solving Sioux Falls network.

t_1	0.0600	t_{20}	0.0460	t_{39}	0.0775	t_{58}	0.0808
t_2	0.0402	t_{21}	0.1270	t_{40}	0.1180	t_{59}	0.1036
t_3	0.0600	t_{22}	0.1011	t_{41}	0.1295	t_{60}	0.0461
t_4	0.0884	t_{23}	0.0935	t_{42}	0.0889	t_{61}	0.1046
t_5	0.0402	t_{24}	0.1284	t_{43}	0.1127	t_{62}	0.0925
t_6	0.0486	t_{25}	0.0506	t_{44}	0.1304	t_{63}	0.0924
t_7	0.0411	t_{26}	0.0552	t_{45}	0.0512	t_{64}	0.0923
t_8	0.0484	t_{27}	0.1286	t_{46}	0.0933	t_{65}	0.0499
t_9	0.0281	t_{28}	0.1120	t_{47}	0.1053	t_{66}	0.1248
t_{10}	0.0847	t_{29}	0.1247	t_{48}	0.1301	t_{67}	0.0930
t_{11}	0.0280	t_{30}	0.1776	t_{49}	0.1016	t_{68}	0.0924
t_{12}	0.1087	t_{31}	0.0854	t_{50}	0.0348	t_{69}	0.0495
t_{13}	0.0974	t_{32}	0.1277	t_{51}	0.1775	t_{70}	0.1335
t_{14}	0.0947	t_{33}	0.1337	t_{52}	0.1023	t_{71}	0.0912
t_{15}	0.1169	t_{34}	0.1139	t_{53}	0.0802	t_{72}	0.1311
t_{16}	0.0588	t_{35}	0.0411	t_{54}	0.0210	t_{73}	0.0493
t_{17}	0.0559	t_{36}	0.1328	t_{55}	0.0347	t_{74}	0.0812
t_{18}	0.0210	t_{37}	0.0310	t_{56}	0.0461	t_{75}	0.1256
t_{19}	0.0457	t_{38}	0.0310	t_{57}	0.0512	t_{76}	0.0497

Table 11: The resulting equilibrium link flows from solving Sioux Falls network.

x_1	6.6117	x_{20}	13.9081	x_{39}	16.2801	x_{58}	10.2387
x_2	9.5834	x_{21}	5.8748	x_{40}	9.0660	x_{59}	8.9514
x_3	6.8855	x_{22}	8.1800	x_{41}	9.3401	x_{60}	23.5386
x_4	7.4205	x_{23}	15.3377	x_{42}	8.4098	x_{61}	8.9850
x_5	9.3097	x_{24}	5.9425	x_{43}	21.1721	x_{62}	6.9501
x_6	18.6187	x_{25}	24.4434	x_{44}	9.4240	x_{63}	7.8222
x_7	15.2883	x_{26}	24.3975	x_{45}	18.2917	x_{64}	6.9563
x_8	18.4954	x_{27}	18.1153	x_{46}	17.7595	x_{65}	9.3722
x_9	22.6678	x_{28}	21.1013	x_{47}	8.3726	x_{66}	10.6043
x_{10}	6.3151	x_{29}	15.6432	x_{48}	15.4477	x_{67}	17.7493
x_{11}	22.6053	x_{30}	8.4342	x_{49}	12.0336	x_{68}	7.8211
x_{12}	9.1783	x_{31}	6.3642	x_{50}	19.9374	x_{69}	9.3819
x_{13}	15.6613	x_{32}	18.0856	x_{51}	8.4327	x_{70}	9.6804
x_{14}	7.6942	x_{33}	8.4341	x_{52}	12.0139	x_{71}	8.4095
x_{15}	9.4394	x_{34}	8.9825	x_{53}	10.2598	x_{72}	9.6788
x_{16}	18.9802	x_{35}	15.1379	x_{54}	17.8759	x_{73}	8.9006
x_{17}	14.1828	x_{36}	8.4800	x_{55}	19.8542	x_{74}	16.1756
x_{18}	17.6012	x_{37}	17.9425	x_{56}	23.4572	x_{75}	10.6008
x_{19}	19.5151	x_{38}	17.9481	x_{57}	18.3465	x_{76}	8.8986

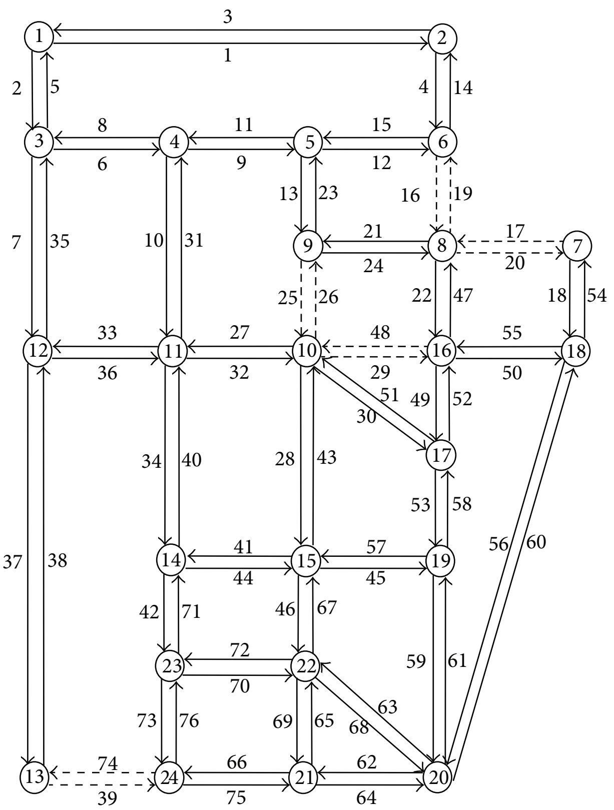

Figure 3: Sioux Falls network.

References

[1] J. G. Wardrop, "Some theoretical aspects of road traffic research," *Proceedings of the Institution of Civil Engineers Part II*, vol. 1, pp. 325–378, 1952.

[2] O. Baskan, "Harmony search algorithm for continuous network design problem with link capacity expansions," *KSCE Journal of Civil Engineering*, pp. 1–11, 2013.

[3] M. Abdulaal and L. J. LeBlanc, "Continuous equilibrium network design models," *Transportation Research B*, vol. 13, no. 1, pp. 19–32, 1979.

[4] C. Suwansirikul, T. L. Friesz, and R. L. Tobin, "Equilibrium decomposed optimisation: a heuristic for the continuous equilibrium network design problem," *Transportation Science*, vol. 21, no. 4, pp. 254–263, 1987.

[5] P. Marcotte, "Network optimization with continuous control parameters," *Transportation Science*, vol. 17, no. 2, pp. 181–197, 1983.

[6] P. Marcotte and G. Marquis, "Efficient implementation of heuristics for the continuous network design problem," *Annals of Operations Research*, vol. 34, no. 1, pp. 163–176, 1992.

[7] T. L. Friesz, R. L. Tobin, H.-J. Cho, and N. J. Mehta, "Sensitivity analysis based heuristic algorithms for mathematical programs with variational inequality constraints," *Mathematical Programming*, vol. 48, no. 2, pp. 265–284, 1990.

[8] H. J. Cho, *analysis of equilibrium network flows and its application to the development of solution methods for equilibrium network design problems [Ph.D. thesis]*, University of Pennsylvania, Philadelphia, Pa, USA, 1988.

[9] Y. Hai, "Sensitivity analysis for queuing equilibrium network flow and its application to traffic control," *Mathematical and Computer Modelling*, vol. 22, no. 4–7, pp. 247–258, 1995.

[10] H. Yang, "Sensitivity analysis for the elastic-demand network equilibrium problem with applications," *Transportation Research B*, vol. 31, no. 1, pp. 55–70, 1997.

[11] T. L. Friesz, H.-J. Cho, N. J. Mehta, R. L. Tobin, and G. Anandalingam, "Simulated annealing approach to the network design problem with variational inequality constraints," *Transportation Science*, vol. 26, no. 1, pp. 18–26, 1992.

[12] T. L. Friesz, G. Anandalingam, N. J. Mehta, K. Nam, S. J. Shah, and R. L. Tobin, "The multiobjective equilibrium network design problem revisited: a simulated annealing approach," *European Journal of Operational Research*, vol. 65, no. 1, pp. 44–57, 1993.

[13] G. A. Davis, "Exact local solution of the continuous network design problem via stochastic user equilibrium assignment," *Transportation Research B*, vol. 28, no. 1, pp. 61–75, 1994.

[14] Q. Meng, H. Yang, and M. G. H. Bell, "An equivalent continuously differentiable model and a locally convergent algorithm for the continuous network design problem," *Transportation Research B*, vol. 35, no. 1, pp. 83–105, 2001.

[15] S.-W. Chiou, "Bilevel programming for the continuous transport network design problem," *Transportation Research B*, vol. 39, no. 4, pp. 361–383, 2005.

[16] X. Ban, H. X. Liu, J. Lu, and M. C. Ferris, "Decomposition scheme for continuous network design problem with asymmetric user equilibria," *Transportation Research Record*, no. 1964, pp. 185–192, 2006.

[17] A. Karoonsoontawong and S. T. Waller, "Dynamic continuous network design problem: linear bilevel programming and metaheuristic approaches," *Transportation Research Record*, no. 1964, pp. 104–117, 2006.

[18] Z. Gao, H. Sun, and H. Zhang, "A globally convergent algorithm for transportation continuous network design problem," *Optimization and Engineering*, vol. 8, no. 3, pp. 241–257, 2007.

[19] T. Xu, H. Wei, and G. Hu, "Study on continuous network design problem using simulated annealing and genetic algorithm," *Expert Systems with Applications*, vol. 36, no. 2, pp. 1322–1328, 2009.

[20] T. V. Mathew and S. Sharma, "Capacity expansion problem for large urban transportation networks," *Journal of Transportation Engineering*, vol. 135, no. 7, pp. 406–415, 2009.

[21] D. Z. W. Wang and H. K. Lo, "Global optimum of the linearized network design problem with equilibrium flows," *Transportation Research B*, vol. 44, no. 4, pp. 482–492, 2010.

[22] C. Li, H. Yang, D. Zhu, and Q. Meng, "A global optimization method for continuous network design problems," *Transportation Research B*, vol. 46, pp. 1144–1158, 2012.

[23] O. Baskan and M. Dell'Orco, "Artificial bee colony algorithm for continuous network design problem with link capacity expansions," in *Proceedings of the 10th International Congress on Advances in Civil Engineering*, Middle East Technical University, Ankara, Turkey, October 2012.

[24] O. Baskan and H. Ceylan, "Modified differential evolution algorithm for the continuous network design problem," in *Proceedings of the 16th Euro Working Group on Transportation*, Porto, Portugal, 2013.

[25] H. Yang and J. Y. T. Wang, "Travel time minimization versus reserve capacity maximization in the network design problem," *Transportation Research Record*, no. 1783, pp. 17–26, 2002.

[26] G. Ziyou and S. Yifan, "A reserve capacity model of optimal signal control with user-equilibrium route choice," *Transportation Research B*, vol. 36, no. 4, pp. 313–323, 2002.

[27] S.-W. Chiou, "A hybrid approach for optimal design of signalized road network," *Applied Mathematical Modelling*, vol. 32, no. 2, pp. 195–207, 2008.

[28] C. S. Fisk, "Optimal signal controls on congested networks," in *Proceedings of the 9th International Symposium on Transportation and Traffic Theory*, pp. 197–216, VNU Science Press, Utrecht, 1984.

[29] M. Frank and P. Wolfe, "An algorithm for quadratic programming," *Naval Research Logistics Quarterly*, vol. 3, pp. 95–110, 1956.

[30] Y. Sheffi, *Urban Transport Networks: Equilibrium Analysis With Mathematical Programming Methods*, Prentice-Hall, New Jersey, NJ, USA, 1985.

[31] X. S. Yang and S. Deb, "Cuckoo search via Lévy flights," in *Proceedings of the world congress on nature and biologically inspired computing (NaBIC '09)*, pp. 210–214, IEEE Press, Piscataway, NJ, USA, December 2009.

[32] X.-S. Yang and S. Deb, "Engineering optimisation by cuckoo search," *International Journal of Mathematical Modelling and Numerical Optimisation*, vol. 1, no. 4, pp. 330–343, 2010.

[33] X.-S. Yang and S. Deb, "Multiobjective cuckoo search for design optimization," *Computers & Operations Research*, vol. 40, no. 6, pp. 1616–1624, 2013.

[34] R. B. Payne, M. D. Sorenson, and K. Klitz, *The Cuckoos*, Oxford University Press, 2005.

[35] E. Valian, S. Mohanna, and S. Tavakoli, "Improved Cuckoo search algorithm for global optimization," *International Journal of Communications and Information Technology*, vol. 1, no. 1, pp. 31–44, 2011.

[36] X. S. Yang, *Nature-Inspired Metaheuristic Algorithms*, Luniver Press, 2nd edition, 2010.

[37] X. S. Yang and S. Deb, "Cuckoo search: recent advances and applications," *Neural Computing and Applications*, 2013.

[38] E. Valian and E. Valian, "A cuckoo search algorithm by Lévy flights for solving reliability redundancy allocation problems," *Engineering Optimization*, 2013.

[39] I. Pavlyukevich, "Lévy flights, non-local search and simulated annealing," *Journal of Computational Physics*, vol. 226, no. 2, pp. 1830–1844, 2007.

[40] S. Walton, O. Hassan, K. Morgan, and M. R. Brown, "Modified cuckoo search: a new gradient free optimisation algorithm," *Chaos, Solitons and Fractals*, vol. 44, no. 9, pp. 710–718, 2011.

[41] A. Kaveh and T. Bakhshpoori, "Optimum design of steel frames using Cuckoo Search algorithm with Lévy flights," *Structural Design of Tall and Special Buildings*, vol. 22, no. 13, pp. 1023–1036, 2013.

[42] R. E. Allsop, "possibilities for using traffic control to influence trip distribution and route choice," in *Proceedings of the 6th International Symposium on Transportation and Traffic Theory*, pp. 345–375, Elsevier, New York, NY, USA, 1974.

[43] H. Yang and S. Yagar, "Traffic assignment and signal control in saturated road networks," *Transportation Research A*, vol. 29, no. 2, pp. 125–139, 1995.

[44] P. Luathep, A. Sumalee, W. H. K. Lam, Z.-C. Li, and H. K. Lo, "Global optimization method for mixed transportation network design problem: a mixed-integer linear programming approach," *Transportation Research B*, vol. 45, no. 5, pp. 808–827, 2011.