



Research article

BO-B&B: A hybrid algorithm based on Bayesian optimization and branch-and-bound for discrete network design problems

Ruyang Yin¹, Jiping Xing^{2,*}, Pengli Mo², Nan Zheng¹ and Zhiyuan Liu²

¹ Institute of Transport Studies, Department of Civil Engineering, Monash University, Clayton, Victoria, Australia

² Jiangsu Key Laboratory of Urban ITS, Jiangsu Province Collaborative Innovation Center of Modern Urban Traffic Technologies, School of Transportation, Southeast University, China

* **Correspondence:** Email: xingjiping@seu.edu.cn.

Abstract: A discrete network design problem (DNDP) is conventionally formulated as an analytical bi-level programming problem to acquire an optimal network design strategy for an existing traffic network. In recent years, multimodal network design problems have benefited from simulation-based models. The nonconvexity and implicitness of bi-level DNDPs make it challenging to obtain an optimal solution, especially for simulation-related models. Bayesian optimization (BO) has been proven to be an effective method for optimizing the costly black-box functions of simulation-based continuous network design problems. However, there are only discrete inputs in DNDPs, which cannot be processed using standard BO algorithms. To address this issue, we develop a hybrid method (BO-B&B) that combines Bayesian optimization and a branch-and-bound algorithm to deal with discrete variables. The proposed algorithm exploits the advantages of the cutting-edge machine-learning parameter-tuning technique and the exact mathematical optimization method, thereby balancing efficiency and accuracy. Our experimental results show that the proposed method outperforms benchmarking discrete optimization heuristics for simulation-based DNDPs in terms of total computational time. Thus, BO-B&B can potentially aid decision makers in mapping practical network design schemes for large-scale networks.

Keywords: Bayesian optimization; branch-and-bound; discrete network design problem; simulation-based optimization

1. Introduction

Network design problems (NDPs) involve determining a design scheme for new transportation network components or existing infrastructure configurations to achieve various social welfare objectives (e.g., minimizing the total network travel cost). The motivation of this study is to consider several existing problems that are associated with finding optimal numbers and locations for new links, apart from expanding the existing ones with budget limitations, while considering the corresponding travel behavior of network users [1,2]. Generally, network design problems can be divided into three categories: discrete network design problems (DNDPs), which improve network capacity by adding new links or lanes; continuous network design problems (CNDPs), which expand the network by increasing the capacity of the existing links; and mixed network design problems (MNDPs), which combine CNDPs and DNDPs [3]. The literature emphasizes that the DNDP is of great academic and practical importance [4–6]. With the growth in population coupled with economic development, the significance of the DNDP increases, as transportation infrastructure cannot deal with the growing travel demand. However, land resources for capacity enhancement remain extremely limited, particularly in metropolitan areas [7]. Accordingly, this study aims to develop a network design strategy that includes the allocation of new links and the number of lanes to be constructed for maximizing the social benefits with limited investments in transportation.

The DNDP can be analyzed from the perspective of optimization as a hierarchical decision-making problem that involves network managers at the upper level, who determine the network design strategy, and network users at the lower level, who pursue minimal individual travel time in response to the given network design. The total system cost is affected by both manager and user decision variables [8–10]. Owing to two groups of stakeholders within the network, the DNDP is typically formulated using bi-level programming models.

In general, the DNDP is most commonly solved using general approaches such as Lagrangian relaxation, decomposition methods and descent methods, after transforming bi-level formulations into single-level ones [11]. These transformations are often based on unrealistic assumptions for simplifying constraints, and this leads to the dynamic traffic being captured inaccurately [12]. Furthermore, these algorithms are computationally demanding for practical purposes [13,14]. Considering their solution efficiency, meta-heuristic algorithms have become increasingly popular in recent decades. Evolutionary algorithms, swarm intelligence and hybrid meta-heuristic algorithms have been used to address DNDPs; these approaches sacrifice the optimality of the solution but significantly reduce the computational time [6,15–17].

Although scholars have made many efforts, the DNDP is still criticized as an old and impractical approach by practitioners because of the limited practical applicability of the existing methods [11]. To deal with real industry-specific projects, it is necessary to acknowledge the existing gaps between engineering and theory. First, traffic models for cities differ from benchmark networks, such as Sioux Falls, in their complexity. Second, large-scale networks are rarely considered in experimental cases. Therefore, the largely simplified multimodal features of actual networks in the literature should be considered.

Recently, advanced simulation packages have provided the potential to generate multimodal environments close to reality and are flexible in constructing networks at varying scales [18]. However, the non-convex and costly to-evaluate relationships between the simulation inputs and outputs make simulation-based optimization extremely challenging because there are no closed-form functions to compute or approximate the derivative.

Inspired by state-of-the-art parameter tuning techniques for machine learning models, Bayesian optimization (BO) has proven to be an effective algorithm for achieving a global optimizer of costly black-box functions for simulation-based optimization problems [19]. The BO method generally comprises two components. First, a surrogate of the objective function is constructed, and the model uncertainty is quantified using a Bayesian-type statistical model (e.g., Gaussian process regression). Further, the location to sample next using an acquisition function derived from the surrogate model is determined [20]. The successful application of BO to a simulation-based CNBP was reported by Yin et al. [21]. Their experimental results suggest that BO has a predominant advantage in reducing the number of simulation runs for simulation-based optimization problems. Standard BO relies on acquisition functions that are defined only on continuous domains, implying that the input variables must be continuous. Therefore, the optimization of DNDPs with discrete inputs using BO is a challenging task that remains unexplored.

Considering the aforementioned literature, this study first develops a simulation-based bi-level model of the DNDP that enables the simulation of a real-world large-scale network and captures accurate multimodal traffic dynamics. Next, to solve the proposed model, a novel hybrid method, BO-B&B, which combines Bayesian optimization and the branch-and-bound algorithm, is developed. By combining state-of-the-art machine-learning parameter-tuning techniques with an exact mathematical optimization technique, the proposed algorithm can achieve both efficiency and accuracy for real-scale DNDPs.

The remainder of this paper is structured as follows: A literature review is introduced in Section 2. The problem description is presented in Section 3. The proposed BO-B&B algorithm is presented in Section 4 to solve a simulation-based multimodal DNDP. In Section 5, a numerical example based on the Sioux-Falls network is presented. Finally, Section 6 concludes the paper.

2. Literature review

To present the innovation of our proposed research model and solution algorithm, this section is divided into the solution of DNDPs and the application of BO-based methods.

2.1. The solution of DNDPs

Previous studies have demonstrated that solving DNDPs can be challenging, even on experimental and small-scale networks [22,23]. Various methods have been used to address DNDPs, including exact mathematical algorithms, heuristics and meta-heuristic algorithms, owing to their practical significance. Leblanc [9] first employed the branch-and-bound (B&B) algorithm to obtain optimal solutions for DNDPs. Poorzahedy and Turnquist [24] proposed a heuristic B&B method to address a single-level formulated DNDP. Farvareh and Sepehri [25] further presented a more efficient B&B algorithm that can compute lower bounds. The difficulty in exploiting the B&B approach lies in its ability to accommodate the inherent nonconvexity of the DNDP resulting from complex equilibrium constraints [4].

To map the relationship between improved flows and newly added links, the concept of the support function was developed by Gao et al. [23]. A generalized Benders decomposition method was employed as the solution algorithm. Using variational inequality constraints (VI) to formulate user equilibrium (UE) conditions, Luathep et al. [26] transformed a bi-level DNDP into a single-level

optimization problem and solved it using a cutting plane algorithm. To provide optimal solutions to a bi-level DNDP, a single-level mixed-integer linear programming (MILP) formulation was developed by Farvaresh and Sepehri [27]. The conventional DNDP was later extended by Wang et al. [5] to a multi-capacity DNDP that determines both the locations and numbers of added lanes of candidate links. To solve the multi-capacity DNDP, two global optimization methods, namely, SO-relaxation and UE-reduction, were developed by exploiting the difference between the system optimization (SO) and UE conditions. Zhang et al. [28] proposed an active-set algorithm to solve a single-level multimodal DNDP with complementarity constraints. Fontaine and Minner [29] reformulated single-level MILP for a DNDP by adopting a piecewise linear approximation and solved it using the Benders decomposition method. Fontaine and Minner [30] further extended the DNDP to solve a dynamic maintenance planning model using the same solution algorithm. Wang et al. [6] developed a novel DNDP that can simultaneously optimize the network design strategy and capacities. The formulated model was solved using global optimization methods that incorporate outer approximation, linearization and range reduction techniques. Bagloee et al. [11] proposed a hybrid method integrating B&B and Benders decomposition for a DNDP considering multiclass and multimodal traffic flows.

2.2. The application of BO-based methods

Previous attempts have been made to develop BO-based methods for other discrete optimization problems [31]. One naïve idea assumes that there is no change in the expected value of the objective function at discrete points and that the inputs can simply be rounded to the nearest integer [32]. For simplicity, this method is referred to as naïve BO in the following text. However, this approach involves a risk in that the BO may become stuck and repeat preexisting observations. Luong et al. [33] attempted to improve the efficiency of naïve BO by adjusting the length scale factor in the kernel function and the tradeoff parameter in the acquisition function. Another approach is sequential model-based optimization for general algorithm configuration (SMAC), in which a random forest surrogate model replaces GP to deal with discrete input [34]. Nevertheless, the random forest has limitations with regard to extrapolation [35]. Another tree-based method, Tree-Parzen estimators (TPEs), can deal with discrete variables by sampling from discrete distributions and estimating the density of high-quality and low-quality candidate points in the search domain [36]. However, the density distribution cannot be efficiently modelled with TPE until there are sufficient observations at the beginning. The disadvantages of the existing approaches pose challenges in the optimization of the discrete black-box functions.

3. Problem description

This section first introduces the notations used and then presents the simulation-based bi-level DNDP model with budget constraints. The notations used in the proposed framework are defined in Table 1.

Table 1. List of notations.

Notation	Description
N	the set of nodes in the network
A	the set of all links in the network, $A = A_1 \cup A_2$
A_1	the set of all existing links in the network
A_2	the set of all proposed new links in the network
\mathbf{u}	the network design plan, i.e., the vector of decision variables, defined as $\mathbf{u} = (u_a, a \in A_2)$
u_a	the number of lanes to be added to link $a \in A_2$
m_a	the maximum number of lanes to be added to link $a \in A_2$
g_a	the construction cost of one lane of new link $a \in A_2$
v_a	the traffic flow on link $a \in A$
$t_a(v_a)$	the link travel time (or cost) of traffic flow v_a on link $a \in A$
I	the total investment budget

The DNDP can be considered as a basic Stackelberg game, in which the transportation network managers are the leaders, and the network users, or followers, react to the leader's decisions through route choices [3]. This interaction can be commonly be represented using a bi-level structure formulation, in which the upper-level objective is to pursue an optimal network design strategy to minimize the total system cost constrained by government budgets, and the lower-level deals with the traffic assignment problem corresponding to the upper-level solution in a user-optimal manner. The upper level of the DNDP can be formulated as follows [3,14,37,38].

Upper Level

$$\min_{\mathbf{u}} Z = \sum_{a \in A_1} v_a t_a(v_a) + \sum_{a \in A_2} v_a t_a(v_a) \quad (1)$$

$$\text{s.t. } \sum_{a \in A_2} g_a u_a \leq I \quad (2)$$

$$0 \leq u_a \leq m_a, \quad \forall a \in A_2 \quad (3)$$

$$u_a \in \mathbb{Z}, \quad \forall a \in A_2 \quad (4)$$

where v_a and $t_a(v_a)$ can be obtained by solving a simulation-based lower-level model. Constraint (2) ensures that the total network improvement cost does not exceed the required investment budget. Constraints (3) and (4) guarantee that the number of newly added lanes is a nonnegative integer and will not exceed the maximum number of lanes that can be accommodated to the potential links. Notably, most existing literature on DNDPs considers the binary restriction of the decision variable u_a , which denotes whether the candidate project link should be added without determining the number of lanes [9,21,23], while only a few studies focus on the number of added lanes [4]. Evidently,

considering the number of lanes directly increases the difficulty of the problem. In this study, the upper-level model is assumed to make optimal decisions regarding the number of newly added lanes to provide clear construction guidance.

A successful traffic network improvement project highly depends on how the outcome is evaluated or, more precisely, on the way it predicts the result of increased network capacity. Thus, it is important to accurately model the route choice of multimodal network users after determining the traffic network design strategy. Conventional analytical models achieve the resultant traffic pattern by solving an equilibrium assignment problem, in which the link travel time is formulated using the Bureau of Public Roads (BPR) functions, and the objective is to achieve minimal individual travel costs [4,39,40]. However, ignoring the impact of signals and multimodal traffic in urban areas is unrealistic for analytical models.

With the aid of simulation-based techniques, the lower level can construct an accurate function to establish a relationship between network user reactions and network design strategies. Transportation networks with signalized intersections and public transport modes can be easily established by exploiting the advantages of off-the-shelf simulation packages such as VISUM simulation software. The network flow pattern $\mathbf{v}(\mathbf{u})=(v_a(\mathbf{u}), a \in A)^T$ can be simulated using an advanced simulation package under approximate equilibrium conditions. To reflect the stochastic nature of travel and traffic, randomness is introduced into the simulation. Every simulation run involves sampling from probability distributions that account for uncertainty, such as the passenger waiting time for public transit. Thus, the simulated link flow pattern can be considered a stochastic variable, denoted by $\Phi^y(\mathbf{u})=(\Phi_a^y(\mathbf{u}), a \in A)^T$. Herein, $\Phi_a^y(\mathbf{u})$ stands for the equilibrium link flow assigned to link $a \in A$ given the network design strategy \mathbf{u} in the simulator, which can be expressed as follows.

Lower Level

$$\mathbf{v}(\mathbf{u}) = E[\Phi^y(\mathbf{u})] \quad (5)$$

where the link flow pattern is represented as the expectation $E[\cdot]$ of the stochastic network performance measures $\Phi^y(\mathbf{u})$, which are evaluated using the embedded traffic assignment module in the simulator.

Without the loss of generality, actuated control strategies are utilized in the signal settings of added lanes, and the operational plans of public transits remain unchanged throughout the optimization process. The Monte Carlo method was adopted to determine the expected link pattern $\mathbf{v}(\mathbf{u})$, considering stochasticity in the simulation run. For a given network design project \mathbf{u} , assume that r independent simulation evaluations, denoted $\Phi_1^y(\mathbf{u}), \Phi_2^y(\mathbf{u}), \dots, \Phi_r^y(\mathbf{u})$, are observed by changing the random seeds in the simulator. Then, $\mathbf{v}(\mathbf{u})$ can be approximated using the sample average as follows:

$$\mathbf{v}(\mathbf{u}) = \frac{1}{r} \sum_{i=1}^r \Phi_i^y(\mathbf{u}), \quad i = 1, 2, \dots, r. \quad (6)$$

where the evaluation batch r is maintained constant across all iterations. The objective function can be evaluated at the upper level once the output link flow pattern is assigned.

Consequently, a complete simulation-based bi-level framework was developed to effectively address the DNDP through iterative interaction between the upper and lower levels.

4. BO-B&B algorithm

In this section, an optimization solution framework combining Bayesian optimization (BO) and the branch-and-bound (B&B) algorithm is presented for simulation-based DNDPs.

4.1. BO-B&B overview

Although simulation-based models flexibly capture the network structure and conveniently solve the multimodal traffic assignment problem, the optimization process is more difficult than in analytical models because there is no explicit function between the simulation input and output. To optimize the objective function with fewer evaluations, an approach that can depict the implicit input-output function surface is required. Therefore, in the proposed solution framework, BO was employed as a technique to evaluate black-box objective functions that are costly to evaluate. Two components constitute Bayesian optimization: a Bayesian statistical model as a surrogate of the objective function, which is a nonparametric Gaussian process regression, and an acquisition function, which is easy to solve [19]. Compared with other data-driven models, Bayesian optimization models do not require a great amount of training data [41], and data-driven models are suitable for handling complex features from multiple categories of datasets [42,43]. Bayesian optimization allocates the remaining evaluations after random sampling using both prior and posterior probability distributions and can consider the tradeoff between exploration and exploitation in unexplored domains.

However, in the conventional Bayesian optimization algorithm, the objective function is assumed to be continuous in all acquisition functions so that they return real-valued points in the next evaluation. Given the discrete optimization problem, where only discrete inputs are allowed, incorporating the B&B algorithm can provide an optimal integer solution of the acquisition function. The B&B algorithm iteratively builds a tree structure consisting of nodes connected by directed branches. For each node in the tree, a “partial solution” to the integer programming problem is identified. By performing these operations, the B&B algorithm ensures that the global optima can be determined by recursively searching the tree of instances formed by the branch operation.

The following subsections provide an illustration of the key components of the proposed hybrid algorithm (BO-B&B), followed by the pseudocode and a step-by-step procedure description.

4.2. Gaussian process

Here, we explain how a Gaussian process is used to build the relationship between a set of sample points \mathbf{x} and the unknown objective function, $f(\mathbf{x})$. Rasmussen and Williams [44] provide a more detailed illustration.

By assessing any sample point \mathbf{x}_i using a predetermined mean function μ_0 , a mean vector can be constructed for the sample set. Generally, a constant value is selected as the mean function:

$$\mu_0(\mathbf{x}) = \mu. \quad (7)$$

A kernel function κ_0 is introduced for the two samples (\mathbf{x}_i and \mathbf{x}_j), and we build a covariance matrix based on the kernel function. For data pairs with a larger positive correlation, their kernel function values are more similar to those of pairs that are far apart.

The radial basis function (RBF), also known as the Gaussian kernel, is one of the most commonly used kernels.

$$\kappa_0(\mathbf{x}_i, \mathbf{x}_j) = \alpha^2 \exp\left(-\frac{\|\mathbf{x}_i, \mathbf{x}_j\|^2}{2d^2}\right) \quad (8)$$

where $\|\mathbf{x}_i, \mathbf{x}_j\|^2$ is the Euclidean distance between the two samples $(\mathbf{x}_i, \mathbf{x}_j)$, α is a variable parameter of the kernel, and d is defined as the length scale of the kernel. Liu et al. [41] found that when the RBF kernel is used, severe overfitting occurs if the objective function is too complex and overly flexible; this phenomenon is termed as the deformation of GP regression. Because our simulation-based DNDP is a fully “black-box” problem and has an unknown functional form, a more generalized kernel function, the Matérn kernel, which can avoid such deformations, is selected. The Matérn kernel is expressed as follows:

$$\kappa_0(\mathbf{x}_i, \mathbf{x}_j) = \alpha^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\sqrt{2\nu} \frac{\|\mathbf{x}_i, \mathbf{x}_j\|}{d}\right)^\nu K_\nu\left(\sqrt{2\nu} \frac{\|\mathbf{x}_i, \mathbf{x}_j\|}{d}\right) \quad (9)$$

where K_ν is a modified Bessel function, Γ denotes a gamma function, and ν controls the degree of kernel smoothness, which is nonnegative. For $\nu \rightarrow \infty$, the Matérn kernel converges to the RBF.

The resultant prior distribution on $f(\mathbf{X}) = [f(\mathbf{x}_1), f(\mathbf{x}_2), \dots, f(\mathbf{x}_n)]$ becomes

$$f(\mathbf{X}) \sim N(\boldsymbol{\mu}_0(\mathbf{X}), \boldsymbol{\kappa}_0(\mathbf{X}, \mathbf{X})). \quad (10)$$

When a new sample \mathbf{x}_{n+1} is added to the set, the prior distribution is updated as follows:

$$\begin{bmatrix} f(\mathbf{X}) \\ f(\mathbf{x}_{n+1}) \end{bmatrix} \sim N\left(\begin{bmatrix} \boldsymbol{\mu}_0(\mathbf{X}) \\ \boldsymbol{\mu}_0(\mathbf{x}_{n+1}) \end{bmatrix}, \begin{bmatrix} \boldsymbol{\kappa}_0(\mathbf{X}, \mathbf{X}) & \boldsymbol{\kappa}_0(\mathbf{X}, \mathbf{x}_{n+1}) \\ \boldsymbol{\kappa}_0(\mathbf{X}, \mathbf{x}_{n+1})^T & \boldsymbol{\kappa}_0(\mathbf{x}_{n+1}, \mathbf{x}_{n+1}) \end{bmatrix}\right). \quad (11)$$

Based on Bayes' theorem, the posterior/conditional probability distribution of $f(\mathbf{x}_{n+1})$ is derived as follows:

$$f(\mathbf{x}_{n+1}) | f(\mathbf{X}) \sim N\left(\boldsymbol{\mu}_{n+1}(\mathbf{x}_{n+1}), \sigma_{n+1}^2(\mathbf{x}_{n+1})\right), \quad (12)$$

$$\boldsymbol{\mu}_{n+1}(\mathbf{x}_{n+1}) = \boldsymbol{\kappa}_0(\mathbf{X}, \mathbf{x}_{n+1}) \boldsymbol{\kappa}_0(\mathbf{X}, \mathbf{X})^{-1} (f(\mathbf{X}) - \boldsymbol{\mu}_0(\mathbf{X})) + \boldsymbol{\mu}_0(\mathbf{x}_{n+1}), \quad (13)$$

$$\sigma_{n+1}^2(\mathbf{x}_{n+1}) = \boldsymbol{\kappa}_0(\mathbf{x}_{n+1}, \mathbf{x}_{n+1}) - \boldsymbol{\kappa}_0(\mathbf{x}_{n+1}, \mathbf{X}) \boldsymbol{\kappa}_0(\mathbf{X}, \mathbf{X})^{-1} \boldsymbol{\kappa}_0(\mathbf{X}, \mathbf{x}_{n+1}). \quad (14)$$

4.3. Acquisition functions

To select the location for the next observation, an acquisition function can be constructed using the posterior mean and covariance of the $f(\mathbf{x}_{n+1})$ derived using Eqs (13) and (14). In general,

acquisition functions are inexpensive and simultaneously account for both “exploration” (such as the uncertainty associated with unexplored domains) and “exploitation” (such as the expected performance of the posterior mean). The three most commonly used acquisition functions are 1) expected improvement (EI), 2) probability of improvement (PoI) and 3) upper confidence bound (UCB) [45]. As multiple parameterized acquisition functions are available in the literature, it is often unclear which one should be employed [46]. Among them, the upper confidence bound (UCB) method is arguably the most commonly used, because the trade-off between exploitation and exploration is straightforward and easy to balance [47]. Therefore, without loss of generality, we select UCB as our sample acquisition function in this study, and the mathematical expression of UCB is as follows.

Upper confidence bound (UCB)

The UCB acquisition function combines the explicit exploitation component $\mu(\mathbf{x})$ and exploration component $\sigma(\mathbf{x})$, and the mathematical expression of UCB for minimization problems is as follows:

$$\Psi(\mathbf{x}) = \mu(\mathbf{x}) - \beta\sigma(\mathbf{x}) \quad (15)$$

where $\beta > 0$ is a tradeoff parameter for achieving a balance between high expected performance and high uncertainty.

Using the above acquisition function, the next-best observation is chosen from \mathbf{X}_{n+1} , whose acquisition function value is the smallest.

$$\mathbf{x}_{n+1} = \arg \min[\Psi(\mathbf{x})] \quad (16)$$

Compared with the previous black-box objective function, these acquisition functions have deterministic functional forms that are relatively easy to solve, and they allow for the evaluation of derivatives. With the conventional BO algorithm, Eq (19) can be solved with a first-order or second-order optimization algorithm, such as the quasi-Newton approach [48]. Thus, BO only suggests a continuous point to be used when sampling the next point for function evaluation, leading to an invalid DNDP input. Thus, we incorporate the B&B algorithm to solve the acquisition functions in BO, which ensures a discrete output. In the next subsection, the B&B algorithm incorporated in BO is introduced.

4.4. Branch-and-bound algorithm embedded in BO

After the aforementioned Gaussian process and acquisition function are established, the problem to be solved is as follows:

$$\min_{\mathbf{u}}[\Psi(\mathbf{u})], \quad (17)$$

$$\text{s.t. } \sum_{a \in A_2} g_a u_a \leq I, \quad (18)$$

$$0 \leq u_a \leq m_a, \quad \forall a \in A_2, \quad (19)$$

$$u_a \in \mathbb{Z}, \quad \forall a \in A_2. \quad (20)$$

Denote the formulations (17)–(20) as the original problem P_0 and the region of feasible solutions Ω . Traditional B&B algorithms are described as a query through a search tree, in which the original problem P_0 is represented by a root node, and each node corresponds to a subproblem of that problem. As NP-hard problems have an exponential number of feasible solutions, instance trees containing an exponential number of leaves exist. A bounding function g exists, which is linked to each node in the tree, referred to as the bound of the node. Particularly, for leaves, the bound equals the value of the corresponding solution, while for internal nodes, the value is a lower bound for any solution in the subspace where the node resides. The following three conditions must be satisfied by the bounding function g :

- i. $g(P_i) \leq \Psi(P_i)$ for all nodes P_i in the tree;
- ii. $g(P_i) = \Psi(P_i)$ for all leaves in the tree;
- iii. $g(P_i) \geq g(P_j)$ if P_j is the father of P_i .

This guarantees that g is a bounding function that contains more information about additional constraints for a subproblem, and these bounds are closely aligned with the objective function. During the search process, a search tree is dynamically formed, initially containing only a root node. There are several problems in which a feasible solution can be generated using a heuristic, and the result is used as the current best solution. The B&B algorithm selects nodes for exploration from a pool of live nodes for each iteration. These nodes correspond to the unexplored feasible subproblems. Using eager strategies, a branching process is performed by adding constraints to the subproblem of the node and producing two or more children. Consequently, the subspace is subdivided into smaller subspaces. Using the result of the sub-problem optimization, the bound for the node for each of these is calculated. Once the node corresponds to a feasible solution, or the bound represents the value of an optimal solution, the value of the node is compared with the current best solution, and the optimal solution is retained. If the subproblem cannot achieve any viable solution, it can be viewed as no bound among the current best solution. Subproblems are also considered if no feasible solutions exist. Alternatively, a better solution in the subproblem cannot be ruled out; subsequently, the node that includes the bound information stored becomes a member of the live subproblem pool.

Consequently, the B&B algorithm comprises three major elements: 1) a bounding function, 2) a node selection strategy and 3) a branching rule. Any successful implementation of the B&B algorithm relies heavily on the bounding function, as a poor bounding function cannot be compensated for by good selection or branching strategies. An ideal bounding function should be capable of generating the best feasible solution for a given sub-problem. To construct the bounding function, we relaxed the integer constraints of the original problem, thereby expanding the set of feasible solutions. Sequential quadratic programming, which is the most commonly used method for constrained nonlinear optimization, is used to solve relaxed problems [49].

4.5. Pseudocode of BO-B&B algorithm

The pseudocode of the BO-B&B algorithm is as follows. Notably, the proposed method can solve specific DNDPs and generic discrete optimization problems.

BO-B&B Algorithm

Initialization:

- 1 Input initial sample solution set $\mathbf{X}_{1:n} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$, select the proper kernel function $\kappa_0(\mathbf{x}_i, \mathbf{x}_j)$, mean function $\mu_0(\mathbf{x})$, and acquisition function $\Psi(\mathbf{x})$
- 2 Evaluate $f(\mathbf{x})$ at initial sample solution set $\mathbf{X}_{1:n}$ according to simulation experiment to obtain $f(\mathbf{x}_{1:n}) = \{f(\mathbf{x}_1), f(\mathbf{x}_2), \dots, f(\mathbf{x}_n)\}$
- 3 set $N = n$

Main Loop:

- 4 Place a Gaussian process prior on $f(\mathbf{x})$ based on sample set $D_{1:N} = \{[\mathbf{x}_1, \mathbf{f}(\mathbf{x}_1)], [\mathbf{x}_2, \mathbf{f}(\mathbf{x}_2)], \dots, [\mathbf{x}_N, \mathbf{f}(\mathbf{x}_N)]\}$
- 5 While $N < N_{MAX}$ do:
 - 6 Update the posterior probability distribution $f(\mathbf{x}_{n+1}) | f(\mathbf{X}_{1:N}) \sim N(\boldsymbol{\mu}_{n+1}(\mathbf{x}_{n+1}), \sigma_{n+1}^2(\mathbf{x}_{n+1}))$ using all available data, Construct acquisition function $\Psi(\mathbf{x}_{n+1})$
 - 7 Define the $\min_{\mathbf{x}_{n+1}} [\Psi(\mathbf{x}_{n+1})]$ with budget constraints, variable bound constraints and integer constraints as original problem P_0

Embedded B&B:*Initialization:*

- 9 Set a sub-problem list $L = \{P\}$
 - 10 Find a feasible integer solution of P_0 using heuristics **, denote the relaxed solution as \mathbf{x}_U
Initialize the best solution $\mathbf{x}^* = \mathbf{x}_U$
 - 11
 - 12 Initialize the upper bound $UB = f(\mathbf{x}_U)$
 - Inner loop:*
 - 13 while $L \neq \emptyset$, do
 - 14 Select a sub-problem P^i from L based on node selection rules
 - 15 Remove P^i from L
 - 16 Find optimal solution of P^i with integer restrictions relaxed, denote the relaxed solution as \mathbf{x}_R
 - 17 If $f(\mathbf{x}_R) \geq UB$,
 - 18 do nothing
 - 19 Elseif $\mathbf{x}_R \in \mathbf{Z}^k$,
 - 20 $UB = f(\mathbf{x}_R), \mathbf{x}^* = \mathbf{x}_R$
 - 21 Else
 - 22 Select a variable $x_R^i \notin \mathbf{Z}, i \in \{1, \dots, k\}$ based on the branching rule
 - 23 Partition problem P^i into two new sub-problem $P_1^i = P^i \cap x_R^i \leq \lfloor x_R^i \rfloor$
and $P_2^i = P^i \cap x_R^i \geq \lceil x_R^i \rceil$
 - 24 End if
 - 25 Set $L = L - \{P^i, P_1^i, P_2^i\}$
 - 26 End while
 - 27 $\mathbf{x}_{n+1} = \mathbf{x}^*$
 - 28 Evaluate $f(\mathbf{x}_{n+1})$ based on the simulation experiment results
 - 29 set $N = n+1$
 - 30 End while
 - 31 Return a solution: the point evaluated with the minimum objective function value
-

** The heuristic denotes the sequential quadratic programming method in this study.

4.6. Step-by-step solution framework

This subsection provides a detailed solution process for specifically simulation-based DNDPs using BO-B&B.

Step 1. Initialization

A multimodal transport network is established in the simulation package, and the loop counter is set as $m=0$.

Step 2. Sample construction

A small sample set of network design strategies, $\mathbf{U}=[\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_n]$, is generated. The samples were selected using a random sampling method.

Step 3. Multimodal traffic assignment in simulator

For each \mathbf{u}_i , the added links are constructed in the selected macroscopic simulator (e.g., VISUM). The traffic private vehicle and public transit assignment procedures are executed under approximated equilibrium conditions in the simulator. The link flow pattern and total network travel time under the network design strategy \mathbf{u}_i is generated.

Step 4. Objective function value calculation

Given the outputs of the simulation in Step 3, we can evaluate the function $Z(\mathbf{U})=[Z(\mathbf{u}_1), Z(\mathbf{u}_2), \dots, Z(\mathbf{u}_n)]$ using Eq (1).

Step 5. Gaussian process regression

With the observed samples, the posterior probability distribution of $Z(\mathbf{u}_{n+1})$ is modeled using GP. A surrogate model mapping the relationship between the discrete network design \mathbf{u} and total travel time $Z(\mathbf{u})$ is established.

Step 6. Next sample point determination

With the posterior mean and covariance of $Z(\mathbf{u}_{n+1})$ calculated in Step 5, we constructed the acquisition function $\Psi(\mathbf{u}_{n+1})$. Using the B&B algorithm, we find the optimal \mathbf{u}_{n+1} which has the best potential to optimize our acquisition function. The selected \mathbf{u}_{n+1} is stored, and we subsequently update the iteration counter $n=n+1$.

Step 7. Check stop criterion

The procedure terminates if both the following criteria are satisfied: 1) The number of iterations exceeds the maximum number N of iterations; 2) the permitted relative gap δ between the two iterations is less than 10^{-6} . The stop criteria are the maximum number of iterations and the permitted relative gap δ between two simultaneous iterations. If $n > N$ and $\delta < 10^{-6}$, the iteration is terminated, and the network design strategy \mathbf{u} that minimizes the objective function is returned. Otherwise, the algorithm returns to Step 3, and \mathbf{u}_{n+1} is evaluated in the simulator. The newly determined sample point is then applied to the sample set for GP updating.

5. Numerical experiment

The Sioux Falls network, which constitutes 24 nodes and 76 links, is shown in Figure 1. This network has been widely studied in transportation experiments. The source data on traffic demands are listed in Table A1, which originated from a research paper published by Wang et al. [5]. Since multimodal traffic is currently everywhere in urban areas, in our research, the traffic network is extended to a multimodal environment in accordance with reality, which allows autos, buses, rails and combined modes to be considered.

Five projects were selected as potential lane addition links, and detailed information is shown in

Table 2. Candidate projects were assumed to be two-way links, and the variables corresponded to the number of new lanes. In actual complex network scenarios or under restricted land planning, our proposed model and solution algorithm can also be applied to both scenarios with existing candidate links and without. In existing studies on DNDPs, we found that the feasible region is generally very small. To better present the algorithm performance, we provide detailed network design guidance in which the transport authority considers adding 0, 1, 2, 3 or 4 lanes to these links. This setting results in up to $5^5=3,125$ total feasible design solutions. Without loss of generality, actuated control strategies were utilized for the signal settings of both existing links and added lanes, and the operational plans of public transits remain unchanged throughout the optimization process. The cost of adding one new lane was set as one unit of the budget, and a total of 10 investment budgets were assumed. In other words, no more than 10 lanes can be added to the network to improve capacity.

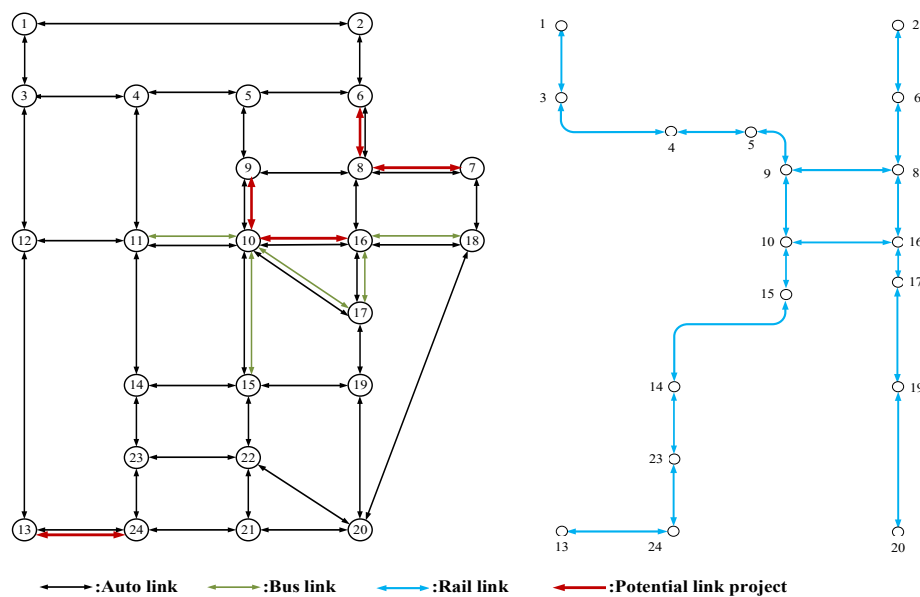


Figure 1. Sioux Falls Network.

The proposed BO-B&B method was compared with existing machine learning methods for discrete optimization, such as BO with naive rounding (Naive BO), sequential model-based optimization for general algorithm configuration (SMAC), Tree-Parzen estimators (TPE) and random search (RS). These baselines are described in the Introduction section. Figure 2 shows the iteration process of the proposed BO-B&B algorithm and the benchmarks. The results show that BO-B&B generally obtained a better solution in much less computational time than the other algorithms.

Table 2. Proposed link improvement project.

Project number	Links (From node, End node)
1	(6,8) and (8,6)
2	(9,10) and (10,9)
3	(13,24) and (24,13)
4	(10,16) and (16,10)
5	(7,8) and (8,7)

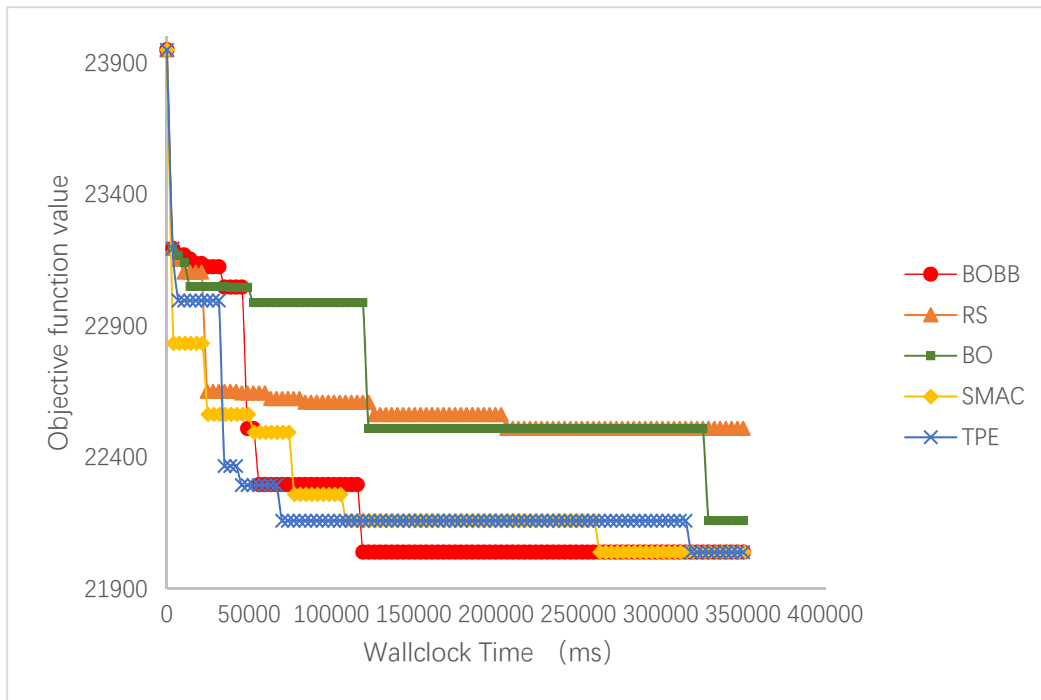


Figure 2. Iteration process of BO-B&B and benchmarks.

Table 3 presents a detailed analysis of the algorithm performance. The result of CPU time is the average of three runs. It can be observed that BO-B&B, TPE and SMAC reached their optimal solutions, but BO-B&B required fewer iterations. Within a limited number of iterations, naive BO and RS failed to find the optimal solution. In terms of reducing the number of iterations, the proposed BO-B&B exhibits predominant advantages. In the optimization process, the efficiency of the optimization process is strongly influenced by the number of simulation runs because each simulation run takes approximately 3 s, which is much longer than the run-time of the algorithm. In other words, the simulations have a substantial effect on the computation time with the increase in the number of simulations. BO-B&B consumes more time for its B&B operations than other algorithms; however, compared with simulations, the amount of extra computation is almost negligible. It should be noted that no particular optimization algorithm is likely to outperform all others in every case. It is not suitable to use BO-based algorithms for problems with more than 20 dimensions because of the complex matrix operations involved.

Table 3. Comparison of algorithm performances.

Algorithm	RS	SMAC	TPE	Naive BO	BO-B&B
Algorithm CPU time (ms) in one iteration	20.3	523.7	52.3	48.9	210.7
CPU time (s) to find optimal solution	/	263.02	318.55	/	119.132

The optimal discrete network design obtained for the five potential projects is (3, 0, 4, 3, 0), implying that projects 1, 3, and 4 should add three, four and three lanes, respectively. The optimized discrete network design induces a 1907 h reduction in the network travel cost. Assuming the average

value of time to be \$15/h, which is a reference from Jayasooriya et al. [50], the network improvement is suggested to save at least \$10.44 million annually, representing a significant reduction in terms of operational savings. Consequently, the optimal discrete network design scheme can be considered as an option to improve the performance of multimodal urban transportation networks.

6. Conclusions

We reformulated the bi-level DNDP using a simulation-based method. The objective of the upper layer of the proposed model is to optimize network design, providing the minimal total system cost subject to budget constraints. Lower-level traffic assignment in a multimodal traffic network was performed using simulation-related methods. We used VISUM simulation software, an advanced simulation package, to evaluate each discrete network design plan/solution, which provides the system performance. To solve the simulation-based bi-level optimization problem, an algorithm called BO-B&B, which combines Bayesian optimization and branch-and-bound, is proposed. This combination provides a suitable approach for solving discrete optimization problems by BO-based methods. Furthermore, Bayesian optimization can rely on previous empirical values and accelerate to obtain the next optimal values in the fewest iterations, which is suitable for problems that are “expensive to evaluate,” even lacking known special structures like concavity or linearity.

According to the results, BO-B&B produced more systematic savings in computational budgets than the three other machine learning techniques. Using BO-B&B, the number of simulation evaluations and iterations required to determine the global optimum of a simulation-based optimization problem was significantly reduced. Using simulation-related methods, we can easily develop complex transport networks with combined modes and signal controls. In this regard, the proposed model may be a useful tool for network managers to deal with a variety of discrete decision variables in the context of real-world distribution network development plans.

Based on our current research, the following investigations will be performed: First, the BO-B&B algorithm will be further modified and extended to high-dimensional problems using parallel computing. Second, for some key areas in the network, mesoscopic and microscopic simulation models will be combined to refine the results.

Acknowledgments

This study is supported by the Distinguished Young Scholar Project (No. 71922007) and Key Project (No. 52131203) of the National Natural Science Foundation of China.

Conflict of interest

The authors declare there is no conflict of interest.

References

1. R. Z. Farahani, E. Miandoabchi, W. Y. Szeto, H. Rashidi, A review of urban transportation network design problems, *Eur. J. Oper. Res.*, **229** (2013), 281–302. <https://doi.org/10.1016/j.ejor.2013.01.001>

2. E. Chen, Z. Ye, C. Wang, M. Xu, Subway passenger flow prediction for special events using smart card data, *IEEE Trans. Intell. Transp. Syst.*, **21** (2019), 1109–1120. <https://doi.org/10.1109/TITS.2019.2902405>
3. H. Yang, M. G. H. Bell, Models and algorithms for road network design: a review and some new developments, *Transp. Rev.*, **18** (1998), 257–278. <https://doi.org/10.1080/01441649808717016>
4. S. Wang, Q. Meng, H. Yang, Global optimization methods for the discrete network design problem, *Transp. Res. Part B Methodol.*, **50** (2013), 42–60. <https://doi.org/10.1016/j.trb.2013.01.006>
5. D. Z. Wang, H. Liu, W. Y. Szeto, A novel discrete network design problem formulation and its global optimization solution algorithm, *Transp. Res. Part E Logist. Transp. Rev.*, **79** (2015), 213–230. <https://doi.org/10.1016/j.tre.2015.04.005>
6. G. Wang, Z. Gao, M. Xu, Integrating link-based discrete credit charging scheme into discrete network design problem, *Eur. J. Oper. Res.*, **272** (2019), 176–187. <https://doi.org/10.1016/j.ejor.2018.05.069>
7. D. Huang, J. Xing, Z. Liu, Q. An, A multi-stage stochastic optimization approach to the stop-skipping and bus lane reservation schemes, *Transportmetrica A: Transp. Sci.*, **17** (2021), 1272–1304. <https://doi.org/10.1080/23249935.2020.1858206>
8. D. Huang, Y. Wang, S. Jia, Z. Liu, S. Wang, A Lagrangian relaxation approach for the electric bus charging scheduling optimization problem, *Transportmetrica A: Transp. Sci.*, **2022** (2022), 1–24. <https://doi.org/10.1080/23249935.2021.2023690>
9. L. J. Leblanc, An algorithm for the discrete network design problem, *Transp. Sci.*, **9** (1975), 183–199. <https://doi.org/10.1287/trsc.9.3.183>
10. Q. Cheng, Y. Chen, Z. Liu, A bi-level programming model for the optimal lane reservation problem, *Expert Syst. Appl.*, **189** (2022), 116147. <https://doi.org/10.1016/j.eswa.2021.116147>
11. S. A. Bagloee, M. Sarvi, M. Patriksson, A hybrid branch-and-bound and benders decomposition algorithm for the network design problem, *Comput. Aided Civ. Infrastruct. Eng.*, **32** (2017), 319–343. <https://doi.org/10.1111/mice.12224>
12. Q. Cheng, Z. Liu, Y. Lin, X. S. Zhou, An s-shaped three-parameter (S3) traffic stream model with consistent car following relationship, *Transp. Res. Part B Methodol.*, **153** (2021), 246–271. <https://doi.org/10.1016/j.trb.2021.09.004>
13. Q. Cheng, Z. Liu, W. Y. Szeto, A cell-based dynamic congestion pricing scheme considering travel distance and time delay, *Transportmetrica B: Transp. Dyn.*, **7** (2019), 1286–1304. <https://doi.org/10.1080/21680566.2019.1602487>
14. Q. Cheng, Z. Liu, J. Guo, X. Wu, R. Pendyala, B. Belezamo, et al., Estimating key traffic state parameters through parsimonious spatial queue models, *Transp. Res. Part C Emerging Technol.*, **137** (2022), 103596. <https://doi.org/10.1016/j.trc.2022.103596>
15. C. Iliopoulou, K. Kepaptsoglou, E. Vlahogianni, Metaheuristics for the transit route network design problem: a review and comparative analysis, *Public Transp.*, **11** (2019), 487–521. <https://doi.org/10.1007/s12469-019-00211-2>
16. H. Poorzahedy, O. M. Rouhani, Hybrid meta-heuristic algorithms for solving network design problem, *Eur. J. Oper. Res.*, **182** (2007), 578–596. <https://doi.org/10.1016/j.ejor.2006.07.038>
17. L. Ahmed, C. Mumford, A. Kheiri, Solving urban transit route design problem using selection hyper-heuristics, *Eur. J. Oper. Res.*, **274** (2019), 545–559. <https://doi.org/10.1016/j.ejor.2018.10.022>

18. Y. Shi, Z. Liu, Z. Wang, J. Ye, W. Tong, Z. Liu, An integrated traffic and vehicle co-simulation testing framework for connected and autonomous vehicles, *IEEE Intell. Transp. Syst. Mag.*, **2022** (2022), 2–15. <https://doi.org/10.1109/MITS.2022.3188566>
19. P. I. Frazier, A tutorial on Bayesian optimization, preprint, arXiv:1807.02811.
20. L. Du, R. Gao, P. N. Suganthan, D. Z. Wang, Bayesian optimization based dynamic ensemble for time series forecasting, *Inf. Sci.*, **591** (2022), 155–175. <https://doi.org/10.1016/j.ins.2022.01.010>
21. R. Yin, Z. Liu, N. Zheng, A simulation-based model for continuous network design problem using Bayesian optimization, *IEEE Trans. Intell. Transp. Syst.*, **2022** (2022), 1–16. <https://doi.org/10.1109/TITS.2022.3176918>
22. L. Yang, Z. Di, M. M. Dessouky, Z. Gao, J. Shi, Collaborative optimization of last-train timetables with accessibility: A space-time network design based approach, *Transp. Res. Part C Emerging Technol.*, **114** (2020), 572–597. <https://doi.org/10.1016/j.trc.2020.02.022>
23. Z. Gao, J. Wu, H. Sun, Solution algorithm for the bi-level discrete network design problem, *Transp. Res. Part B Methodol.*, **39** (2005), 479–495. <https://doi.org/10.1016/j.trb.2004.06.004>
24. H. Poorzahedy, M. A. Turnquist, Approximate algorithms for the discrete network design problem, *Transp. Res. Part B Methodol.*, **16** (1982), 45–55. [https://doi.org/10.1016/0191-2615\(82\)90040-6](https://doi.org/10.1016/0191-2615(82)90040-6)
25. H. Farvaresh, M. M. Sepehri, A branch and bound algorithm for bi-level discrete network design problem, *Netw. Spat. Econ.*, **13** (2013), 67–106. <https://doi.org/10.1007/s11067-012-9173-3>
26. P. Luathep, A. Sumalee, W. H. Lam, Z. C. Li, H. K. Lo, Global optimization method for mixed transportation network design problem: a mixed-integer linear programming approach, *Transp. Res. Part B Methodol.*, **45** (2011), 808–827. <https://doi.org/10.1016/j.trb.2011.02.002>
27. H. Farvaresh, M. M. Sepehri, A single-level mixed integer linear formulation for a bi-level discrete network design problem, *Transp. Res. Part E Logist. Transp. Rev.*, **47** (2011), 623–40. <https://doi.org/10.1016/j.tre.2011.02.001>
28. L. Zhang, H. Yang, D. Wu, D. Wang, Solving a discrete multimodal transportation network design problem, *Transp. Res. Part C Emerging Technol.*, **49** (2014), 73–86. <https://doi.org/10.1016/j.trc.2014.10.008>
29. P. Fontaine, S. Minner, Benders decomposition for discrete-continuous linear bilevel problems with application to traffic network design, *Transp. Res. Part B Methodol.*, **70** (2014), 163–172. <https://doi.org/10.1016/j.trb.2014.09.007>
30. P. Fontaine, S. Minner, A dynamic discrete network design problem for maintenance planning in traffic networks, *Ann. Oper. Res.*, **253** (2017), 757–772. <https://doi.org/10.1007/s10479-016-2171-y>
31. S. Ding, Z. Wang, J. Zhang, F. Han, X. Gu, Time delay system identification using controlled recurrent neural network and discrete Bayesian optimization, *Appl. Intell.*, **52** (2022), 8351–8371. <https://doi.org/10.1007/s10489-021-02823-3>
32. E. C. Garrido-Merchán, D. Hernández-Lobato, Dealing with categorical and integer-valued variables in Bayesian optimization with gaussian processes, *Neurocomputing*, **380** (2020), 20–35. <https://doi.org/10.1016/j.neucom.2019.11.004>
33. P. Luong, S. Gupta, D. Nguyen, S. Rana, S. Venkatesh, Bayesian optimization with discrete variables, in *Australasian Joint Conference on Artificial Intelligence*, Springer, Cham, (2019), 473–484. https://doi.org/10.1007/978-3-030-35288-2_38

34. F. Hutter, H. H. Hoos, K. Leyton-Brown, Sequential model-based optimization for general algorithm configuration, in *International conference on learning and intelligent optimization*, Springer, Berlin, Heidelberg, (2011), 507–523. https://doi.org/10.1007/978-3-642-25566-3_40
35. B. Lakshminarayanan, D. M. Roy, Y. W. Teh, Mondrian forests for large-scale regression when uncertainty matters, preprint, arXiv:1506.03805.
36. J. Bergstra, D. Yamins, D. D. Cox, Hyperopt: A python library for optimizing the hyperparameters of machine learning algorithms, in *Proceedings of the 12th Python in science conference*, (2013), 1-8. <https://doi.org/10.25080/MAJORA-8B375195-003>
37. D. E. Boyce, Urban transportation network-equilibrium and design models: recent achievements and future prospects, *Environ. Plann. A: Econ. Space*, **16** (1984), 1445–1474. <https://doi.org/10.1068/a161445>
38. T. L. Magnanti, R. T. Wong, Network design and transportation planning: models and algorithms, *Transp. Sci.*, **18** (1984), 1–55. <https://doi.org/10.1287/trsc.18.1.1>
39. J. Huo, X. Wu, C. Lyu, W. Zhang, Z. Liu, Quantify the road link performance and capacity using deep learning models, *IEEE Trans. Intell. Transp. Syst.*, **2022** (2022), 1–11. <https://doi.org/10.1109/TITS.2022.3153397>
40. Z. Chen, X. Li, X. Qu, A continuous model for designing corridor systems with modular autonomous vehicles enabling station-wise docking, *Transp. Sci.*, **56** (2021), 1–30. <https://doi.org/10.1287/trsc.2021.1085>
41. Z. Liu, C. Lyu, J. Huo, S. Wang, J. Chen, Gaussian process regression for transportation system estimation and prediction problems: the Deformation and a Hat Kernel, *IEEE Trans. Intell. Transp. Syst.*, **2022** (2022), 1–12. <https://doi.org/10.1109/TITS.2022.3155527>
42. Y. Liu, F. Wu, C. Lyu, S. Li, J. Ye, X. Qu, Deep dispatching: A deep reinforcement learning approach for vehicle dispatching on online ride-hailing platform, *Transp. Res. Part E Logist. Transp. Rev.*, **161** (2022), 102694. <https://doi.org/10.1016/j.tre.2022.102694>
43. Y. Liu, R. Jia, J. Ye, X. Qu, How machine learning informs ride-hailing services: A survey, *Commun. Transp. Res.*, **2** (2022), 100075. <https://doi.org/10.1016/j.commtr.2022.100075>
44. C. Rasmussen, C. Williams, *Gaussian Processes for Machine Learning*, MIT Press, Cambridge, MA, USA, 2006. <https://doi.org/10.7551/mitpress/3206.001.0001>
45. T. Pourmohamad, H. K. Lee, Bayesian optimization via barrier functions, *J. Comput. Graphical Stat.*, **31** (2022), 74–83. <https://doi.org/10.1080/10618600.2021.1935270>
46. E. Brochu, V. M. Cora, N. De Freitas, A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning, preprint, arXiv:1012.2599.
47. B. Hao, Y. Abbasi-Yadkori, Z. Wen, G. Cheng, Bootstrapping upper confidence bound, preprint, arXiv:1906.05247.
48. S. Liu, A. B. Owen, Quasi-monte carlo quasi-newton in variational bayes, *J. Mach. Learn. Res.*, **22** (2021), 11043–11065
49. P. E. Gill, E. Wong, Sequential quadratic programming methods, in *Mixed Integer Nonlinear Programming*, Academic press, (2012), 147–224. https://doi.org/10.1007/978-1-4614-1927-3_6
50. S. A. C. S. Jayasooriya, Y. M. M. S. Bandara, Measuring the Economic costs of traffic congestion, in *2017 Moratuwa Engineering Research Conference (MERCon)*, IEEE, Moratuwa, Sri Lanka, (2017), 141–146. <https://doi.org/10.1109/MERCon.2017.7980471>

Appendix

Table A1. OD travel demand of Sioux Falls network.

O	D	Demand	O	D	Demand	O	D	Demand	O	D	Demand
1	2	100	4	7	400	7	4	400	9	24	200
1	3	100	4	8	700	7	5	200	10	1	1300
1	4	500	4	9	700	7	6	400	10	2	600
1	5	200	4	10	1200	7	8	1000	10	3	300
1	6	300	4	11	1500	7	9	600	10	4	1200
1	7	500	4	12	600	7	10	1900	10	5	1000
1	8	800	4	13	600	7	11	500	10	6	800
1	9	500	4	14	500	7	12	700	10	7	1900
1	10	1300	4	15	500	7	13	400	10	8	1600
1	11	500	4	16	800	7	14	200	10	9	2800
1	12	200	4	17	500	7	15	500	10	11	3900
1	13	500	4	18	100	7	16	1400	10	12	2000
1	14	300	4	19	200	7	17	1000	10	13	1900
1	15	500	4	20	300	7	18	200	10	14	2100
1	16	500	4	21	200	7	19	400	10	15	4000
1	17	400	4	22	400	7	20	500	10	16	4400
1	18	100	4	23	500	7	21	200	10	17	3900
1	19	300	4	24	200	7	22	500	10	18	700
1	20	300	5	1	200	7	23	200	10	19	1800
1	21	100	5	2	100	7	24	100	10	20	2500
1	22	400	5	3	100	8	1	800	10	21	1200
1	23	300	5	4	500	8	2	400	10	22	2600
1	24	100	5	6	200	8	3	200	10	23	1800
2	1	100	5	7	200	8	4	700	10	24	800
2	3	100	5	8	500	8	5	500	11	1	500
2	4	200	5	9	800	8	6	800	11	2	200
2	5	100	5	10	1000	8	7	1000	11	3	300
2	6	400	5	11	500	8	9	800	11	4	1400
2	7	200	5	12	200	8	10	1600	11	5	500
2	8	400	5	13	200	8	11	800	11	6	400
2	9	200	5	14	100	8	12	600	11	7	500
2	10	600	5	15	200	8	13	600	11	8	800
2	11	200	5	16	500	8	14	400	11	9	1400
2	12	100	5	17	200	8	15	600	11	10	4000
2	13	300	5	19	100	8	16	2200	11	12	1400
2	14	100	5	20	100	8	17	1400	11	13	1000
2	15	100	5	21	100	8	18	300	11	14	1600
2	16	400	5	22	200	8	19	700	11	15	1400
2	17	200	5	23	100	8	20	900	11	16	1400
2	19	100	6	1	300	8	21	400	11	17	1000

Continued on next page

O	D	Demand	O	D	Demand	O	D	Demand	O	D	Demand
2	20	100	6	2	400	8	22	500	11	18	200
2	22	100	6	3	300	8	23	300	11	19	400
3	1	100	6	4	400	8	24	200	11	20	600
3	2	100	6	5	200	9	1	500	11	21	400
3	4	200	6	7	400	9	2	200	11	22	1100
3	5	100	6	8	800	9	3	100	11	23	1300
3	6	300	6	9	400	9	4	700	11	24	600
3	7	100	6	10	800	9	5	800	12	1	200
3	8	200	6	11	400	9	6	400	12	2	100
3	9	100	6	12	200	9	7	600	12	3	200
3	10	300	6	13	200	9	8	800	12	4	600
3	11	300	6	14	100	9	10	2800	12	5	200
3	12	200	6	15	200	9	11	1400	12	6	200
3	13	100	6	16	900	9	12	600	12	7	700
3	14	100	6	17	500	9	13	600	12	8	600
3	15	100	6	18	100	9	14	600	12	9	600
3	16	200	6	19	200	9	15	1000	12	10	2000
3	17	100	6	20	300	9	16	1400	12	11	1400
3	22	100	6	21	100	9	17	900	12	13	1300
3	23	100	6	22	200	9	18	200	12	14	700
4	1	500	6	23	100	9	19	400	12	15	700
4	2	200	6	24	100	9	20	600	12	16	700
4	3	200	7	1	500	9	21	300	12	17	600
4	5	500	7	2	200	9	22	700	12	18	200
4	6	400	7	3	100	9	23	500	12	19	300
12	20	500	15	18	200	18	19	300	21	22	1800
12	21	300	15	19	800	18	20	400	21	23	700
12	22	700	15	20	1100	18	21	100	21	24	500
12	23	700	15	21	800	18	22	300	22	1	400
12	24	500	15	22	2600	18	23	100	22	2	100
13	1	500	15	23	1000	19	1	300	22	3	100
13	2	300	15	24	400	19	2	100	22	4	400
13	3	100	16	1	500	19	4	200	22	5	200
13	4	600	16	2	400	19	5	100	22	6	200
13	5	200	16	3	200	19	6	200	22	7	500
13	6	200	16	4	800	19	7	400	22	8	500
13	7	400	16	5	500	19	8	700	22	9	700
13	8	600	16	6	900	19	9	400	22	10	2600
13	9	600	16	7	1400	19	10	1800	22	11	1100
13	10	1900	16	8	2200	19	11	400	22	12	700
13	11	1000	16	9	1400	19	12	300	22	13	1300
13	12	1300	16	10	4400	19	13	300	22	14	1200
13	14	600	16	11	1400	19	14	300	22	15	2600
13	15	700	16	12	700	19	15	800	22	16	1200

Continued on next page

O	D	Demand	O	D	Demand	O	D	Demand	O	D	Demand
13	16	600	16	13	600	19	16	1300	22	17	1700
13	17	500	16	14	700	19	17	1700	22	18	300
13	18	100	16	15	1200	19	18	300	22	19	1200
13	19	300	16	17	2800	19	20	1200	22	20	2400
13	20	600	16	18	500	19	21	400	22	21	1800
13	21	600	16	19	1300	19	22	1200	22	23	2100
13	22	1300	16	20	1600	19	23	300	22	24	1100
13	23	800	16	21	600	19	24	100	23	1	300
13	24	700	16	22	1200	20	1	300	23	3	100
14	1	300	16	23	500	20	2	100	23	4	500
14	2	100	16	24	300	20	4	300	23	5	100
14	3	100	17	1	400	20	5	100	23	6	100
14	4	500	17	2	200	20	6	300	23	7	200
14	5	100	17	3	100	20	7	500	23	8	300
14	6	100	17	4	500	20	8	900	23	9	500
14	7	200	17	5	200	20	9	600	23	10	1800
14	8	400	17	6	500	20	10	2500	23	11	1300
14	9	600	17	7	1000	20	11	600	23	12	700
14	10	2100	17	8	1400	20	12	400	23	13	800
14	11	1600	17	9	900	20	13	600	23	14	1100
14	12	700	17	10	3900	20	14	500	23	15	1000
14	13	600	17	11	1000	20	15	1100	23	16	500
14	15	1300	17	12	600	20	16	1600	23	17	600
14	16	700	17	13	500	20	17	1700	23	18	100
14	17	700	17	14	700	20	18	400	23	19	300
14	18	100	17	15	1500	20	19	1200	23	20	700
14	19	300	17	16	2800	20	21	1200	23	21	700
14	20	500	17	18	600	20	22	2400	23	22	2100
14	21	400	17	19	1700	20	23	700	23	24	700
14	22	1200	17	20	1700	20	24	400	24	1	100
14	23	1100	17	21	600	21	1	100	24	4	200
14	24	400	17	22	1700	21	4	200	24	6	100
15	1	500	17	23	600	21	5	100	24	7	100
15	2	100	17	24	300	21	6	100	24	8	200
15	3	100	18	1	100	21	7	200	24	9	200
15	4	500	18	4	100	21	8	400	24	10	800
15	5	200	18	6	100	21	9	300	24	11	600
15	6	200	18	7	200	21	10	1200	24	12	500
15	7	500	18	8	300	21	11	400	24	13	800
15	8	600	18	9	200	21	12	300	24	14	400
15	9	900	18	10	700	21	13	600	24	15	400
15	10	4000	18	11	100	21	14	400	24	16	300
15	11	1400	18	12	200	21	15	800	24	17	300
15	12	700	18	13	100	21	16	600	24	19	100

Continued on next page

O	D	Demand	O	D	Demand	O	D	Demand	O	D	Demand
15	13	700	18	14	100	21	17	600	24	20	400
15	14	1300	18	15	200	21	18	100	24	21	500
15	16	1200	18	16	500	21	19	400	24	22	1100
15	17	1500	18	17	600	21	20	1200	24	23	700



AIMS Press

©2022 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>).